New statistical learning methods for chemical toxicity data analysis

by
Chaeryon Kang

A dissertation submitted to the faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Biostatistics.

Chapel Hill
2011

Approved by:

Michael R. Kosorok

Yufeng Liu

Fred A. Wright

Hao Zhu

Fei Zou

Abstract

New statistical learning methods for chemical toxicity data analysis.
(Under the direction of Michael R. Kosorok.)

In the first part of the dissertation, we introduce the "change-line" classification and regression method to study latent subgroups. The proposed method finds a line which optimally divides a feature space into two heterogeneous subgroups, each of which yields a response having a different probability distribution or having a different regression model. The procedure is useful for classifying biochemicals on the basis of toxicity, where the feature space consists of chemical descriptors and the response is toxicity activity. In this setting, the goal is to identify subgroups of chemicals with different toxicity profiles. The split-line algorithm is utilized to reduce computational complexity. A two step estimation procedure, using either least squares or maximum likelihood for implementation, is described. Two sets of simulation studies and a data analysis applying our method to rat acute toxicity data are presented to demonstrate utility of the proposed method.

Second, the asymptotic properties in the change-line regression model are studied through empirical process techniques, including consistency and the rates of convergence of M-estimators in the change-line regression model. We prove that the estimator of the regression parameters achieves $\sqrt{n}$-consistency while the estimator of the change-line parameters are n-consistent.

Last, we introduce the Interactive Decision Committee method for classification when high-dimensional feature variables are grouped into feature categories. The proposed method uses the interactive relationships among feature categories to build base classifiers which are combined using decision committees. The proposed procedure is useful for classifying biochemicals on the basis of toxicity activity, where the feature space consists of chemical descriptors belonging to at least one feature category, and the responses are binary indicators of toxicity activity. The support vector machine, random forests, and tree-based AdaBoost algorithms are utilized as classifier inducers. To combine base classifiers, the voting method with forward selection given the number of base classifiers by 5-fold CV and a stacked generalization with two different learning algorithms were utilized. We applied the proposed method to two chemical toxicity data sets. For these data sets, the proposed method improved the classification performance with respect to the average prediction accuracy compared to a single classifier.

Acknowledgments

It is a pleasure to thank many people who made this thesis possible.

First and foremost, I would like to express my sincere appreciation and gratitude to my advisor, Dr. Michael Rene Kosorok, for his support, financial assistance, and constant encouragement. Throughout my dissertation research period, he provided invaluable suggestions and good teaching. He made me believe that theory and application are not separate, rather, theory helps us to solve practical problem better.

I gratefully thank the members of my committee. I thank Dr. Yufeng Liu for his advice and help for computational problems in this research. I gratefully thank Dr. Fred A. Wright for constant encouragement and insightful comments on this thesis. My sincere thanks go to Dr. Hao Zhu for providing me the chemical toxicity data sets and helping me understand them better. Many thanks go in particular to Dr. Fei Zou for her invaluable comments and advice for this thesis.

My special thanks go to Dr. Donglin Zeng. He constantly encouraged me in my first and second years in UNC at Chapel Hill as my academic advisor, and patiently answered my questions about BIOS 760.

I would like to thank Dr. Jianwen Cai (Clinical and Translational Science Award),

Dr. Robert M. Hammer, Dr. John H. Gilmore (UNC Schizophrenia Research Center) and NC Center for Childrens Health Improvement for an excellent opportunity to work with them as a graduate research assistant (GRA) student. I also thank Dr. Chirayath M. Suchindran for his support and advice. I would like to acknowledge the advice and help of Dr. Gary Koch, Dr. Amy Herring, Dr. Bahjat Qaqish and Dr. Todd Schwartz for my GRA research. I also would like to thank Kosorok Research Group (KRG) and Statistical Learning and High-Dimensional Data (SLHDD) working group for helpful discussion.

Last, I would like to thank UNC Biostatistics department staffs, my friends, and my family for all their love and support.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

In medicinal chemistry and toxicology, the assessment of potential toxicity associated with drugs and commercial chemicals is an important topic. Some of the chemicals may be hazardous to human health or to the environment. Standard toxicity assessment requires in vivo testing in animals, which is expensive, time consuming, and raises ethical concerns. For these reasons, only a small fraction of commercial chemicals have been tested extensively. Thus, there is increasing interest in developing models for accurate toxicity prediction, to better prioritize chemicals for testing, with an ultimate goal of purely computational toxicity prediction. Quantitative Structure-Activity Relationship (QSAR) modeling is one of the most popular approaches to develop computational toxicity models [Richard, 2006]. QSAR approaches model the relationship between chemical structures and target biological activities and the resulting models are used to predict the target biological activities using the chemical descriptors of new compounds.

Due to the high-dimensionality of the descriptors, many QSAR models have been developed using learning techniques from machine learning. Recent rapid progress in biomedical research and biotechnology have led to an explosive growth in both

the size and complexity of biomedical data. These biomedical data contain rich information about complex human diseases and biological processes that allows us to better understand disease mechanisms and to provide better treatment to patients. The problem is how to extract underlying information contained in such massive amounts of data. Statistical learning is widely used to resolve statistical problems that we face when we use high-dimensional and complex data. According to Hastie et al. [2009], "statistical learning can be broadly described as a statistical approach to extract important patterns and trends and understand 'what the data says.' We call this learning from the data." Statistical learning can provide a useful tool not only to answer specific research questions but also to make discoveries that formulate new hypothesis. Statistical learning problems can be categorized as either supervised learning (for example, classification or prediction of outcomes) or unsupervised learning (for example, clustering observations based on characteristic features). In this dissertation, we study two specific problems: heterogeneous latent subgroups in a population and the low-prediction accuracy of classification models. We propose two new statistical learning procedures to study these two problems, and apply those methods to improve QSAR models of animal toxicity.

1.1 The change-line method

It is scientifically reasonable to expect the relationship between toxicity activity and chemical descriptors to depend on some latent structure within the population. For example, there may be two groups of chemicals with normally distributed response (toxicity activity) values, one group with large mean and small variance and another group with small mean and large variance. In investigating such associations, it is also possible that a group of chemicals shows a positive relationship between toxicity

activity and a set of descriptors while another group of chemicals shows a negative relationship with the same descriptors. In such cases, we cannot identify the important latent structure using traditional classification and regression methods.

To address this latent classification and regression problem, we consider first the case in which the sample size is n, the feature space is $p-$dimensional, and there are only two latent groups separated by a $p-$dimensional hyperplane. Even in this apparently simple case, the computational complexity is prohibitive, even with moderate to small sample sizes. For this reason, we focus in this preliminary paper on the very simple setting where the feature space is two-dimensional and the latent subgroups of interest are divided by a line which we call the change-line. This simple initial implementation will pave the way for the more general, higher dimensional methodology. When each group has a different probability distribution in the response, we call the approach "change-line classification" and focus on finding the line that divides the complete population into two subgroups and estimates the distribution parameters of the two groups. When each group has a different association between the response variable and a set of covariates (some of which may be also contained in the feature space), we call the approach "change-line regression" and focus on finding a line about which the regression parameters differ.

In some aspects, our approach is similar to change-point analysis, except that the change variable X is two-dimensional rather than one-dimensional. When X is continuous and one-dimensional, a change-point is the location γ on X where the probability distribution of a response Y, or the association between Y and a set of covariates Z, changes for X below γ versus X above γ. In the statistics literature,

there have been numerous studies on the change-point problem in parametric, non-parametric, semiparametric, and Bayesian settings. However, when we consider the change-line setting where $X \in \mathbb{R}^2$, significant computational challenges arise compared to the one-dimensional change-point problem. There have been a few studies considering the change problem in the two-dimensional case, but the approaches were not applicable to the general two-dimensional setting. The method we propose, in contrast, enables a general extension of the one-dimensional change-point problem to the change-line problem in a two-dimensional feature space.

When the response variable is binary, e.g., $Y \in \{1, -1\}$, the problem can be roughly viewed as "latent classification" from machine learning. However, one can apply the latent classification method only when the response variable is binary (or at least categorical). Compared to the classical "latent classification" problem, our approach is more applicable to a broader range of response possibilities in the sense that it is applicable to both continuous and discrete response variables. Moreover, the proposed method combines some ideas from machine learning with statistical modeling. Thus, we can potentially reduce the non-reproducibility problem of the pure machine learning method using principled statistical inference and modeling methodology.

An alternative approach is to apply project pursuit regression (PPR, Friedman and Stuetzle [1981]) or support vector regression (SVR, Vapnik et al. [1996]) to analyze these types of data. Compared to PPR or SVR, the proposed change-line approach can more easily handle the setting where the feature space variables X differ from the regression variables Z. This appears to be impossible for SVR. On the other hand, PPR could potentially be generalized to allow changing coefficients in a regression on Z, where the change is a linear function of the feature space X. We do not pursue

this avenue further here.

A potential application for the change-line approach is development of personalized medicine. Recently, there has been great interest in developing personalized medicine in clinical trials. Some recent studies have shown that people can respond to the same treatment in different ways due to individual characteristics. For example, a recent study by Glynn et al. [2009] for the National Cancer Institute (NCI) found that a genetic variation located in the SOD2 gene is associated with different responses to the chemotherapy drug cyclophosphamide, which is used to treat breast cancer. Our method would be useful in studying heterogeneous subgroups of patients who respond to the treatments differently.

1.2 The interactive decision method

External prediction accuracy is one of the most important issues in QSAR modeling. However, most currently available QSAR toxicity models have relatively low prediction ability for new compounds (Stouch et al. [2003]; Johnson [2008]). The decision committee method, sometimes called *ensemble* or *classifier fusion*, is known to perform better than a single classifier because it associates multiple base classifiers to work as a committee (Opitz and Maclin [1999]; Assareh et al. [2008]). Many recent studies of chemical toxicity have utilized decision committee methods to develop QSAR models. When feature variables belong to some informative categories, each category yields different predictions of outcome due to fundamental differences in the information contained in the variables. Eventually, this increases the diversity among base classifiers. Each category might provide important insight into the data structure by itself (the univariate method) or via association with other categories (the

interactive method). It is scientifically reasonable to assume that different feature categories may be interactively associated, and that such relationships could affect the classification task.

In this dissertation, we propose the interactive decision committee (IDC) method to improve prediction accuracy in binary classification problems when high-dimensional feature variables are grouped into feature categories. The method uses the interactive relationships between existing feature categories to build base classifiers in the decision committee context. This is our first contribution. Our second contribution is to utilize a two-stage 5-fold cross-validation (CV) technique to choose the number of base classifiers to be combined. This technique reduces problems on overtraining by controlling the size of the decision committee method. In addition, we applied stacked generalization to learn a combination rule. We found that this procedure can improve classification performance compared to a single large, unaggregated classifier.

1.3 Outline of thesis

The rest of this dissertation is organized as follows. In Chapter 2, we provide a summary of the literature review of the existing approaches to solving similar problems to the one described above. This includes an introduction of an efficient algorithm to consider all possible lines generated by two points in a two dimensional feature space. For the IDC method, a general setting for the decision committee method and aggregation rules, stacked generalizations, and a brief introduction to Support Vector Machines, Random forests, and AdaBoost algorithms are provided. In Chapter 3, we describe a model and method for the change-line classification and regression

problem, and the feasibility of our method is illustrated with some simulation studies. We also provide a numerical example of the application of our methodology to chemical toxicity data. At the end of this chapter, a result on the development of a hypothesis test for the presence of a change-line is presented. In Chapter 4, we focus on the asymptotic properties of the estimator in the change-line regression model. The consistency and the rates of convergence of M-estimators in the change-line regression model have been studied through empirical process techniques. Next, the IDC method is introduce in Chapter 5. We provide simulation studies and numerical results on the chemical toxicity data in this chapter. We then end this dissertation in Chapter 6 with a discussion of some of the limitations of the proposed methods and further research topics to pursue. Some important results from empirical process theory for the change-line method and additional details on empirical processes and proofs are provided in Chapter 7.

Chapter 2

Background

2.1 Change-line method

2.1.1 Projection pursuit regression

The basic idea of projection pursuit (PP) was originally introduced by Kruskal (Kruskal [1969]; Kruskal [1972]), and then successfully implemented by Friedman and Tukey [Friedman and Tukey, 1974]. The PP algorithm seeks to find interesting linear projections of high dimensional multivariate data by numerically maximizing a given objective function. Working in the low dimensional subspace, PP helps to overcome "the curse of dimensionality" caused by the sparsity of high dimensional samples. Moreover, PP allows one to ignore variables with no or little information, which is an advantage in comparison to local averaging methods [Huber, 1985]. A good reference on the diverse research on PP is Huber [1985]. Friedman and Stuetzle extended the idea of PP to a classification problem [Friedman and Stuetzle, 1980], a regression problem (projection pursuit regression, PPR, [Friedman and Stuetzle, 1981]), and a density estimation problem [Friedman et al., 1984]. Although there is a great deal of interesting literature on projection pursuit classification, for example, Lee et al. [2005], we only provide an overview of the literature on PPR in this section. In some

aspects, the change-line classification and regression method is similar to PPR. PPR is a type of nonparametric smoothing method, and it approximates the regression surface by a sum of general smooth functions of linear combinations of the predictors in an iterative way.

Let $\{X, Y\}_{i=1}^{n}$ be a pair of random variables such that X is a random vector in $\mathbb{R}^p$, and Y is a response variable in $\mathbb{R}$. Let ω_m, $m = 1, 2, \ldots, M$ be a unit p-vector of unknown parameters. Then, the PPR model can be defined as $f(X) = \sum_{m=1}^{M} g_m(\omega_m^T X)$, where $g_m(\omega^T X)$ is a ridge function, $\omega_m^T s$ are projection direction vectors, and M denotes the number of projections. M can be determined in various ways such as cross-validation or a forward stage-wise strategy that stops adding terms when the newly added term does not significantly improve the model. Basically, PPR finds the g_m and ω_m that minimize the objective function

$$\sum_{i=1}^{n}[r_i - \sum_{m=1}^{M} g_m(\omega_m^T x_i)]^2, \tag{2.1}$$

where r_i is a residual, and it is initialized to y_i. Let us consider the simplest case of one-dimensional PPR ($M = 1$). PPR can be accomplished in a two-step procedure. For the first stage, given the direction vector ω and $v = \{v_i\} = \{\omega^T x_i\}_{i=1}^{n}$, we construct a smooth representation $g(v)$ of the current residual as ordered in ascending values of v. It is usually obtained by a one-dimensional smoothing method such as local regression or the smoothing spline method so as to minimize (2.1). Next, ω is obtained by the use of a quasi-newton method as follows: let $\hat{\omega}^{(k)}$ be the estimate of ω at the k^{th} iteration. By the quasi-newton method, we can write $g(\hat{\omega}^{(k+1)T} x_i) \approx g(\hat{\omega}^{(k)T} x_i)$ $+g'(\hat{\omega}^{(k)T} x_i)(\hat{\omega}^{(k+1)} - \hat{\omega}^{(k)})^T x_i$, which gives a new form of objective function (2.1) such

as

$$\sum_{i=1}^{n}[r_i - g(\hat{\omega}^{(k+1)T}x_i)]^2 \approx \sum_{i=1}^{n} g'(\hat{\omega}^{(k)T}x_i)^2[\hat{\omega}^{(k)T}x_i + \frac{r_i - g(\hat{\omega}^{(k)T}x_i)}{g'(\hat{\omega}^{(k)T})x_i} - \hat{\omega}^{(k+1)T}x_i]^2.$$

Thus, $\hat{\omega}^{(k+1)}$ can be estimated by weighted least squares regression with new a response variable $\hat{\omega}^{(k)T}x_i + \frac{r_i - g(\hat{\omega}^{(k)T}x_i)}{g'(\hat{\omega}^{(k)T}x_i)}$ on the x_i with weights $g'(\hat{\omega}^{(k)T}x_i)^2$ without intercept. Once we find the direction vector $\hat{\omega}^{(k+1)}$ and $\hat{v}_i^{(k+1)} = \hat{\omega}^{(k+1)T}x_i$, we can compute the corresponding $g(\hat{v}^{(k+1)T})$. This process is repeated until the improvement in (2.1) is not significant compared to a user-defined threshold.

Huber [1985] and Hastie et al. [2009] discussed some shortcomings of PPR. One of the difficulties of PPR is interpretation of the fitted model. The results of one-dimensional PPR can be understandable as similar to linear regression, but it is not easy to interpret how a single term of projection affects the approximation in higher dimensional PPR ($M \geq 2$). For this reason, PPR is useful for prediction rather than modeling of data except for one-dimensional PPR. Also, since PPR is a linear dimension reducer, it works poorly in the case where a highly nonlinear structure exists in the data [Huber, 1985]. The demanding computational problem could be solved by newly developed efficient algorithms, but PPR is still a computationally expensive method. Although PPR itself is not widely used to solve statistical problems, it affects the development of new methodologies such as neural networks from machine learning and independent component analysis (ICA) which is a very popular method of analyzing brain images [Hastie et al., 2009].

2.1.2 Latent classification/Latent class regression

The change-line classification and regression model could be thought as a *"binary latent variable"* model. That is, the distribution of the observations depends on a binary latent variable, and this variable is completely determined by a line. Extensive literature exists on the latent class model or latent variable model: the following are two articles of particular interest.

Recently, Langseth and Nielsen [2005] developed the "latent classification model (LCM)", which is a family of classifiers for a categorical response variable and a set of continuous attributes. LCM combines a naive Bayes model with a mixture of factor analysis (FA). FA reduces the dimensionality of the attributes space based on the covariance structure of the data. As a result, LCM produces a relatively small model, and it utilizes a set of latent variables to model correlations between attributes, which is not allowed in the classical naive Bayes method. Let Y be a categorical response variable, X be a set of continuous attributes, and Z be a set of latent variables which is also continuous. Conditionally on $Y = j$, the latent variable Z is assumed to follow a Gaussian distribution and to be conditionally independent. Similarly, conditional on $Z = z$, X is assumed to follow a Gaussian distribution independently. Langseth and Nielsen [2005] showed that whenever $X| \ Y = j$ follows a Gaussian distribution, the joint distribution (Y, X) can be represented by an LCM. To learn the LCM classifier, the authors proposed an algorithm to score a model based on its accuracy, which is estimated by using a wrapper approach. Also, an Expectation-Maximization (EM)-algorithm was utilized to learn the model parameter in the Gaussian distribution for X and Z. In a simulation study for comparison of the classification accuracy with other existing methods such as naive Bayes, the k-nearest neighbor method (kNN),

and principle component analysis (PCA), the proposed LCM showed the best performance, and it produced a simpler model. In a later paper, Langseth and Nielsen [2009] extended the LCM to binary LCM by allowing binary attributes with continuous latent variables.

Guo et al. [2006] presented a regression model where both the categorical outcome and the continuous predictors are latent. For individual i, one is interested in the relationship between a $q-$dimensional continuous variable f_i and a categorical variable c_i with K categories. However, neither f_i nor c_i is directly observable. Instead, we observe a $p-$dimensional continuous vector X and $j-$dimensional binary response vector Y_i to measure f_i and c_i, respectively. In this article, the latent variables (f_i, c_i) were assumed to be missing data, and two numerical methods were utilized to obtain maximum likelihood estimators: the Monte-Carlo Expectation and Maximization (MCEM) algorithm and the Gaussian quadrature approximation with a quasi-newton algorithm. Similar to the latent classification models, Y_i is assumed to be conditionally independent of X_i given $c_i = c$ and $f_i = f$. This implies that Y_i and X_i are related only through the relationship of c_i and f_i. Additionally, the latent factor f_i is assumed to follow a normal distribution in order to use the Gaussian quadrature approach.

Both the LCM and latent class regression methods were developed based on the assumption that the latent variable is continuous, and the probability distribution of the latent variable is known. This is different from the change-line approach because we are interested in studying binary latent variables without any distributional assumption of a latent variable.

Table 2.1: Interaction Tree

	$x \le c : t_L$	$x > c : t_R$
Group 1	$\mu_1^L, \bar{y}_1^L, s_1^2, n_1$	$\mu_1^R, \bar{y}_1^R, s_2^2, , n_2$
Group 2	$\mu_2^L, \bar{y}_2^L, s_3^2, n_3$	$\mu_2^R, \bar{y}_2^R, s_4^2, n_4$

2.1.3 Subgroup analysis in machine learning

In subgroup analysis, one is interested in finding subgroups in a population that are sufficiently large and have statistically unusual characteristics related to a property of interest. For example, in comparing treatment effects, researchers are interested in the heterogeneity of the treatment effect across subgroups as well as the overall treatment effect. The traditional approach to subgroup analysis is based on testing interaction effects between treatment and a covariate of interest. When there exists a significant interaction effect, one evaluates the treatment effect within each subgroup. This approach is simple, but the potential subgroups tend to be subjective.

To avoid such subjectivity, Su et al. [2009] proposed a new subgroup analysis approach that combines the idea of recursive partitioning and an interaction tree (IT) procedure. Suppose we have n independently and identically distributed observations $\{y_i, z_i, x_i\}$, where y_i is a continuous response variable, z_i is a binary response variable for two treatments, and $x_i = (x_{i1}, \ldots, x_{ip})^T$ is a $p-$dimensional covariate vector. For a continuous variable x and a threshold value c, observations satisfying $x \le c$ go to the left child node t_L while others go to the right child node t_R. Let $\{\mu_1^L, \bar{y}_1^L, s_1^2, n_1\}$ denote the population mean, sample mean, sample variance, and sample size for group 1 in the node t_L. Similar notation applies to the other quantities as shown in Table (2.1). Then, a statistic to assess the interaction between X and Z is defined by $t(s) = \dfrac{(\bar{y}_1^L - \bar{y}_2^L) - (\bar{y}_1^R - \bar{y}_2^R)}{\hat{\sigma}\sqrt{1/n_1 + 1/n_2 + 1/n_3 + 1/n_4}}$, where a pooled estimator of variance $\hat{\sigma} = \sum_{i=1}^{4} \omega_i s_i^2$,

and $\omega_i = (n_i - 1)/(n - 4)$. For a given split s, $G(s) = t(s)^2$ converges to a $\chi^2(1)$ distribution. Therefore, in the first stage, we seek a best split $s^* = \arg\max_s G(s)$, and grow a tree by repeatedly splitting each node of t_L and t_R. To prune the large tree, this paper utilized an interaction-complexity proposed by Breiman et al. [1984]. Finally, the best size subtree is selected based on the maximum interaction-complexity measure. A simulation study illustrated that the proposed method worked very well both for cases where heterogeneity exists in a population and where it does not. In spite of easy interpretation and good performance, this method has some limitations. It is based on the binary tree method, so the computational complexity of finding a threshold c increases as the dimension and the support of X increase. Also, a separate calculation is required to decide on an interaction-complexity measure, making this method more computationally intensive. In addition, this method can not handle two variables simultaneously to find the optimal split point.

2.1.4 Resampling or subsampling methods

In machine learning, some learning algorithms are too complex, so they require too much time and computation to train large amounts of data. In such cases, training on many random subsamples and averaging them is less time consuming. Similarly, bootstrapping can be used in this situation. Because these alternatives are available, few resampling methods for handling large computational complexity have been studied. Malzahn and Opper [2003] developed a novel method for the approximate calculation of resampling averages in an analytical way. The method combined the replica trick with the Bayesian approach. By avoiding retraining subsamples, this method requires much less computational time than the Monte-Carlo resampling method for the Gaussian process model.

2.1.5 Change-point analysis

Since Page [1954] introduced a change-point problem in the context of quality control, there has been a great deal of literature on change-point analysis in biostatistics, econometrics, and image analysis. One-dimensional change-point analysis has been studied intensively in both parametric and nonparametric ([Muller, 1992], [Wu and Chu, 1993]) approaches, and in both Frequentist and Bayesian ([Chernoff and Zacks, 1964]) approaches. Most literature on change-point analysis has focused on two problems: testing for the existence of a change-point, and estimating and conducting inference regarding the location of the change-point.

One-dimensional change-point analysis

The general form of the model with a change-point in a one-dimensional covariate can be written as follow:

$$h(Y(\theta; X, Z)) \sim C(X;\gamma)F(Z;\beta,\tau) + \{1 - C(X;\gamma)\}G(Z;\delta,\tau),$$

where $C(X;\gamma) = \mathbf{1}\{X > \gamma\}$, $X \in \mathbb{R}$, $Z \in \mathbb{R}^p$, and $\gamma \in [a, b]$. F and G are known, real valued and continuous functions, and h is a link function. To obtain identifiability of the parameters, many studies have assumed that the true value $\beta_0 \neq \delta_0$. To estimate a change-point parameter as well as model parameters, most studies have taken a two-stage approach. First, for a fixed value of the change-point γ, obtain first-stage estimators for model parameters using maximum likelihood (Pons [2003]; Kosorok and Song [2007]) or maximum score estimators (Lee and Seo [2008]). Next, find an estimator for γ that maximizes a given objective function, usually by using the grid method. Some studies have shown that model parameters are adaptive in the sense that the limiting distribution of the model parameters do not depend on

knowledge of the true value of the change-point parameter (Pons [2003]; Kosorok and Song [2007]; Lee and Seo [2008]). This implies that asymptotic confidence intervals for model parameters with unknown change-points are the same as in a model with a change-point at known γ_0. Please see Pons [2003], Kosorok and Song [2007], and Lee and Seo [2008] for more details.

A change-point analysis is frequently applied to the study of the dose-response relationship between a continuous exposure and risk of disease. In a dose-response study, a change point is the unknown level of continuous exposure where the dose-response relationship changes abruptly. Pastor and Guallar [1998] proposed a two-segmented logistic regression method to make inferences on the change-point parameter in the binary response model. Interestingly, a simulation study with a finite sample showed that the maximum likelihood estimator (MLE) for regression parameters depended highly on the MLE for the change-point parameters. When the MLE for the change-point parameter was close to the true value, the MLE for the regression parameter was also close to the true value. Pastor-Barriuso et al. [2003] extended the previous study in Pastor and Guallar [1998] to a more general regression function by allowing both abrupt change and more gradual transition between two different linear trends. The authors introduced a method incorporating a general transition function into the linear logistic regression model. However, both articles pointed out some limitations of the proposed methods due to the assumption of the existence of a change-point and the potential problem of model misspecification.

In a Cox regression model, the assumption of proportional hazards is not always satisfied over the whole range of a covariate, but it may hold within a range of the covariate. To handle this problem, Pons [2003] proposed a two-phase Cox regression

model with a change-point at an unknown threshold in a covariate as follows:

$$\lambda_\theta(t|Z) = \lambda(t)\exp\{\alpha^T Z_1(t) + \beta^T Z_2(t)\mathbf{1}\{Z_3 \le \gamma\} + \delta^T Z_2(t)\mathbf{1}\{Z_3 > \gamma\}\},$$

where λ is a hazard function of a survival time t, $Z = (Z_1, Z_2, Z_3)$ is a vector of a covariate, and γ is a scalar. Several important asymptotic properties of the estimators were obtained based on empirical process methodology. Pons [2003] utilized a maximum partial likelihood estimation method to estimate both regression parameters and a change-point parameter. The Breslow estimator was used to estimate the cumulated hazard function. The author proved that the MLEs of the regression parameters are $\sqrt{n}$–consistent while the MLE for the change-point parameter is n–consistent. Moreover, $n(\hat{\gamma}_n - \gamma_0)$ converges weakly to the value of $\hat{v}_Q$ which is almost surely a finite random time that maximizes a certain right continuous jump process Q. Denoting $\xi = (\alpha, \beta, \delta)$, the limiting distributions of both $\sqrt{n}(\hat{\xi}_n - \xi_0)$ and $\sqrt{n}(\hat{\Lambda}_n - \Lambda_0)$ are asymptotically normal. Also, this paper proved that $\sqrt{n}(\hat{\xi}_n - \xi_0)$ and $n(\hat{\gamma}_n - \gamma_0)$ are asymptotically independent under the some conditions.

Kosorok and Song [2007] generalized Pons's model to a general linear transformation model, and studied a change-point problem occurring at the unknown threshold of a one-dimensional covariate in the transformation models under right censoring as follows:

$$P[T > t|\tilde{Z}(t)] = S_z(t) \equiv \Lambda\Big(\int_0^t e^{\beta^T Z(s) + [\alpha + \eta^T Z_2(s)]\mathbf{1}\{Y > \zeta\}} dA(s)\Big),$$

where α is a scalar, $\zeta \in \mathbb{R}$, $\eta \in \mathbb{R}^q$, and Λ is a known decreasing function with $\Lambda(0) = 1$. The authors proved that the nonparametric maximum likelihood estimator (NPMLE)

of a change-point parameter is an n-consistent estimator while the remaining NPMLEs achieve $\sqrt{n}$-consistency. For the $\sqrt{n}$–consistent estimators, the authors derived a score operator and an information operator through a one-dimensional submodel approach for the infinite-dimensional parameter Λ. This article showed asymptotic normality of the regular parameters while the change-point parameter estimator converges weakly to some maximizer of a right-continuous jump process in some Skorohod space. The authors proposed several Monte-Carlo methods based on bootstrapping to make inferences about the change-point parameter and the regular estimators and for the test of the existence of a change-point. For the test of the existence of a change-point, the authors proposed a supremum score test statistic and a mean score test statistic based on the weighted bootstrapping method. Simulation studies showed that both the supremum and mean score test methods performed well, and the supremum score test method was more powerful then the mean score test.

Lee and Seo [2008] discussed a change-point problem in a threshold binary regression model from a semiparametric point of view based on Manski's maximum score estimator [Manski, 1975]. Suppose we observe a binary outcome Y that is determined by an unobservable continuous variable Y^* by $Y = \mathbf{1}\{Y^* \geq 0\}$. The authors considered a threshold regression model for Y^* as follows:

$$Y^* = \beta_0^T W + \delta_0^T X \mathbf{1}\{D > \gamma_0\} + U,$$

where $\theta_0 = (\beta_0, \delta_0)$, $W \in \mathbb{R}^q$, $Z \in W$, $D \in \mathbb{R}$, and U is an unobserved random variable without imposing any parametric distribution. As Manski showed in his work (Manski [1975]; Manski [1985]), (θ_0, γ_0) are identifiable up to scale under the conditional median independence assumption and some regularity conditions if the distribution of X has sufficiently rich support. Also, it is assumed that $\|\theta_0\| = 1$ and $\gamma_0 \neq 0$, otherwise

γ_0 is not identifiable. Manski's maximum score estimator is defined by

$$S_n(\theta, \gamma) = \sum_{i=1} n(2Y_i - 1)\mathbf{1}\{G(X_i, \theta, \gamma) \geq 0\},$$

where $G(X, \theta, \gamma) = \beta^T w + \delta^T z\mathbf{1}(d > \gamma)$ and $X = (W, Z, d)$. Therefore, they obtain a maximum score estimator $(\hat{\theta}, \hat{\gamma})$ to maximize S_n over the parameter space. This article proved that $\hat{\theta}_n$ has $n^{1/3}-$consistency while $\hat{\gamma}_n$ has $n-$consistency under some assumptions. For the test of existence of a change-point, the authors proposed a hypothesis test based on bootstrap subsampling.

Two-dimensional change-point analysis

There have been a few studies considering the change-point problem in the two-dimensional case. Hartigan [1994] introduced an "adaptive shift estimator" for two-dimensional change-point analysis, but the two-dimensional change-point problem was treated as a product of two one-dimensional change-point problem. Raftery [1994] proposed an approach to estimate change-curves in image based clustering edge pixels that are close to the same curve, but the approach was somewhat "ad-hoc" and not applicable to the general two-dimensional setting. Ninomiya [2004] proposed a method of hypothesis testing for the existence of a change-point in two-dimensional random fields using the tube method. This approach is very interesting, but it is not clear how to incorporate the tube method into the change-point problem due to the threshold being in a two-dimensional covariate. Also, the underlying distribution of the outcome should be assumed to follow a normal distribution (or poisson distribution as in the example in Ninomiya [2004]) to approximate the tail probability. Ninomiya focused on the calculation of the upper bound of the tail probability in a differential geometry context, so it is not clear how to estimate the unknown

parameters.

Convergence rate under model misspecification

Banerjee and McKeague [2007] discussed the problem of the convergence rate of the change-point parameter under a misclassified model with a smooth curve. As shown previously, the convergence rate of the change-point parameter is n when an abrupt change occurs. Under the smooth curve setting, however, the convergence rate of the change-point parameter is $n^{1/3}$, which is much slower than n. This implies that if a working model is misspecified by a smooth curve, then the computed confidence interval for the estimated change-point parameter is misleadingly narrow, and this may result in unstable inferences. In this article, the authors considered the split-point problem in a binary decision tree for a nonparametric regression, and showed that all least squared estimators of model parameters and the split-point converge jointly at a cube-root rate. Therefore, a confidence interval for the split-point can be a good alternative for a confidence interval for the change-point parameter when we may not be sure a priori about the abrupt change. Based on the asymptotic joint distribution of the estimators, this article proposed various ways to construct a confidence interval for a split-point in the binary decision tree such as the subsampling bootstrap, Wald-type, and two statistics based on the residual sum of squares (RSS). According to the simulation study presented in the article, the subsampling bootstrap confidence interval was wider than the other three. The Wald-type confidence interval and the RSS-type confidence intervals showed a severe tendency toward undercoverage.

2.1.6 Split-line algorithm

In this section, we describe the basic idea and the main results of a novel algorithm developed by Kosorok [2008a] which is used for this study. This algorithm considers

all possible lines that can uniquely divide all points in a two-dimensional feature space into two groups in an efficient way regarding computational time and effort.

Let us suppose we have n observations $X_n = (x_1, \ldots, x_n)$, where $x_i \in \mathbb{R}^2$. Consider a map $X \mapsto \omega^T X - \gamma$ for $(\omega, \gamma) \in \mathbb{R}^2 \times \mathbb{R}$. This line $\omega^T X - \gamma$ can divide all points X_n into two groups by defining the classification rule $C(X) = \mathbf{1}\{\omega^T X - \gamma > 0\}$. To achieve parameter identifiability of ω and γ, we assume that $\omega \in \mathcal{S}^2$. Let $R_n(\omega) \equiv \{r_1, \ldots, r_n\}$ be a set of ranks of $\omega^T X_j$, $j = 1, \ldots, n$ for each ω. Let us further suppose that two different ω_i, ω_j, for $i \neq j$, generate classifier $c_i(X) = \mathbf{1}\{\omega_i^T X - \gamma_i > 0\}$ and $c_j(X) = \mathbf{1}\{\omega_j^T X - \gamma_j > 0\}$, respectively. We note that $c_i(X)$ can be identical to $c_j(X)$ up to the value of γ if $R_n(\omega_i) = R_n(\omega_j)$. This is true because we can find a γ_j such that $c_j(X)$ generates the same classifying result as $c_i(X)$ for classifier $c_i(X)$. Therefore, we can reduce the number of all possible lines that divide X_n into two groups if we can remove some subset of the ω's that produce redundant ranks R_n. Fortunately, the following algorithm developed by Kosorok [2008a] gives an efficient way to enumerate $W = \{\omega_1, \ldots, \omega_k\}$, where $k \leq n(n-1)$, that generates all possible unique rankings of $\omega^T X$ for X_n. The basic idea of this algorithm is to consider the angle u_{ij} between the line connected to any two points of $(x_i, x_j) \in \mathbb{R}^2$ and the X-axis, and then express the line as $\omega^T X - \gamma$ where $(\omega_1, \omega_2) = (\cos u, \sin u)$.

Algorithm 1. *(Split-line Algorithm [Kosorok, 2008a])*

1. *For every distinct pair of data points $x_i, x_j \in X_n$, calculate the angle α_{ij} between the line $x_j - x_i$ and the vector $(0, 1)$. This can be done by taking the arctan of the slope of the line $x_j - x_i$.*

2. *Add both the angle $\frac{\pi}{2} + \alpha_{ij}$ and $\frac{3\pi}{2} + \alpha_{ij}$ (subtracting off 2π if the total $> 2\pi$), and save in a set T_n.*

3. *Do (2) for all $1 \le j < i \le n$, to obtain T_n with no more than $n(n-1)$ distinct elements (there may be fewer) all in the range $(0, \pi]$.*

4. *Sort the elements of T_n, and denote the resulting ordered distinct elements $t_1, \ldots, t_k$, where $k \le n(n-1)$.*

5. *Compute*

$$u_j = \begin{cases} \dfrac{(t_j + t_{j+1})}{2}, & \text{for } j = 1, \ldots, k-1 \\ \dfrac{(t_j + t_1 + 2\pi)}{2}, & \text{for } j = k \end{cases}$$

where we take $u_k = u_k - 2\pi$ if $u_k > 2\pi$, and call the resulting collection U.

6. *let W be the set of vectors of the form $(\cos u, \sin u)$ running over $u \in U$.*

7. *To obtain the hyperplanes, iterate through all values of $\omega \in W$, and for each such ω, we only need to check those values of γ that lie between the sorted values of $\omega^T x_1, \ldots, \omega^T x_n$.*

Definition 1. *A set W of ω's has no "redundancy" if $R_n(\omega_1) \ne R_n(\omega_2)$ for $\omega_1 \ne \omega_2$, $\omega_1, \omega_2 \in W$. Also, W is "complete" if for any $\omega \in \mathcal{S}^2$, there exists a $\tilde{\omega} \in W$ such that $R_n(\omega) = R_n(\tilde{\omega})$.*

Theorem 1. *The set W constructed as described above is complete and has no redundancies.*

2.2 The interactive decision committee method

In this section, we provide a brief review of literature on the decision committee method and background methods that were used for this study.

2.2.1 Literature review of the decision committee method

The basic idea of the decision committee method is to integrate multiple base classifiers which are individually trained by a deterministic *inducer* (a mapping from a training sample to a classifier) into the combined classification system. Each base classifier can provide complementary information about the pattern to be classified, which may lead to better performance in the classification task [Vale et al., 2008]. In addition, aggregating multiple predictions from different base classifiers can resolve the problem of overtraining [Shin and Markey, 2006]. Aggregation is the procedure by which multiple classifiers are combined into a single large classifier. A good choice of the aggregation rule can improve the classification accuracy. There are many different aggregation rules in the decision committee method literature (Clemen [1989]; Ali and Pazzani [1996]; Dietterich [1997]). We discuss aggregation rules in detail later in section 2.2.2.

In the decision committee method, *diversity* among base classifiers is one of the key factors to improve classification performance, and can be even more important than the aggregation rule [Assareh et al., 2008]. Lam [2000] and Shipp and Kuncheva [2002] characterized diversity by independence among the base classifiers (independency), tendency to make different decisions (orthogonality), and complementary effects (complementarity) among base classifiers. Krogh and Vedelsby [1995] define diversity as disagreement among the base classifiers on feature variables. It is obvious that there would be no accuracy gained by aggregating multiple classifiers which provide identical information about the classification pattern. Diverse classifiers provide varied information for the classification patterns.

One can increase the diversity among base classifiers through resampling individuals as training sets (for example, boosting [Freund and Schapire, 1997] and bagging [Breiman, 1996]), selecting different subsets of feature variables (for example, the random forest [Breiman, 2001] and the random subspace algorithm [Ho, 1998]), or using different types of learning algorithms to build base classifiers. Recent research has improved classification performance further by integrating boosting or bagging with feature selection (for example, Stefanowski [2005] and Assareh et al. [2008]).

In ensemble feature selection, each base classifier is trained based on different subsets of feature variables. See Opitz [1999], Abeel et al. [2009], Vale et al. [2008], and Tuv et al. [2009] for more details. Many recent studies of chemical toxicity have utilized the ensemble feature selection method to develop QSAR models. Budka and Gabrys [2010] applied a ridge regression ensemble in which base classifiers were trained using different feature subsets selected by the "plus-L-takeaway-R" method [van der Heijden et al., 2004]. Dutta et al. [2007] proposed an ensemble feature selection method to identify an optimal subset of chemical descriptors based on different types of learning algorithms applied simultaneously. Neither study, however, considered the potential interactive relationship between existing categories of descriptors. Including these two studies, most published articles in the decision committee method literature have focused on finding better aggregation rules or on feature selection using marginal prediction ability (Bauer and Kohavi [1999]; Assareh et al. [2008]; Tuv et al. [2009]).

2.2.2 Background of method

To set the stage for our contribution, we briefly introduce the basic idea of the decision committee method and related issues.

General setting for the decision committee method

Suppose we have training data consisting of n pairs $\{(y_i, x_i)\}_{i=1}^n$, where $y_i \in \{-1, 1\}$ is a binary outcome for class level, and $x_i \in \mathbb{R}^p$ is a p-dimensional feature vector. Following the presentation in Kuncheva et al. [2001], we define a *classifier* as a map $C : \mathbb{R}^p \mapsto \{-1, 1\}$. Let $\mu(C(\mathbf{x}))$ denote the output, or class label of C. During the construction phase, multiple base classifiers $C = \{C_1, \ldots, C_L\}$ are trained, and a collection of first-level outputs $\mu(C(x)) = \{\mu(C_1(x)), \ldots, \mu(C_L(x))\}$ are obtained. Then the final class label $\hat{C}$ can be obtained by aggregating the base classifiers through the aggregation rule $\mathcal{F}(C)$ defined in the next section.

Aggregation rules: Voting and stacking

As mentioned in section 2.2.1, one important component of the decision committee method is aggregation of the base classifiers. Researchers have focused on three methods to aggregate base classifiers: selecting the best one (winner takes all), voting for the most popular class, and stacking with other learning algorithm. The winner takes all strategy selects one best base classifier. Voting for the most popular class takes an average over outputs from the base classifiers with or without weighting, and classifies the examples into the class that has the most votes.

Stacking (or stacked generalization), introduced by Wolpert [1992], is a general

method of learning with meta-level classifiers using predictions from the base classifiers as inputs [Sigletos et al., 2005]. The original stacked generalization method can be briefly described as follows. Given a learning set Θ, one partitions data into $\{\Theta_{i1}, \Theta_{i2}\}_{i=1}^{n}$, where each Θ_{i1} consists of $n-1$ observations Θ, and Θ_{i2} is the remaining individual of Θ. This is similar to the leave-one-out cross-validation (LOOCV) technique, but one can use any cross-validation partition set to build sets of $\{\Theta_{i1}, \Theta_{i2}\}$. Suppose there are L different learning algorithms (generalizers), say $G_l^0, l = 1 \ldots, L$. In the level-0 stage, $\{\Theta_{i1}\}_{i=1}^{n}$ are used as level-0 inputs to train G_l^0 level-0 classification inducers. Applying L different learning rules to the remaining $\Theta_2 = \{\Theta_{i2}\}_{i=1}^{n}$ produces level-0 outputs, $\{G_l^0(\Theta_2)\}_{l=1}^{L}$. In the level-1 stage, those L sets of level-0 outputs are used as level-1 inputs for stage-1 learning algorithm G^1, and finally level-1 outputs are obtained.

Wolpert [1992] discussed that stacked generalization can improve prediction by deducing the biases of the base classifiers for a given learning set. Many researchers, including Wolpert [1992] and Ting et al. [1997], have reported that stacked generalization produces better results than voting or selecting the best one in many empirical experiments. In practice, however, some experimental results based on real-world data have shown different results. Ting et al. [1997] showed that the performance of majority voting was better than stacked generalization or selecting the best one in a data set in which the standard error between the error rates of the worst performing level-0 classifier is small. Wolpert [1992] noted that stacked generalization may not perform well when it is applied to small, noisy data sets since reproduction of the learning set may not be achievable. Wolpert [1992] discussed some concerns related to stacked generalization, but we do not address those in this thesis. In this study, we combine base classifiers by using either voting (majority vote) or stacking.

1. Aggregation by voting. We first introduce two voting schemes that were used in this study. Suppose we have outputs $\{\mu(C_1(x)), \ldots, \mu(C_L(x))\}$, where $\mu(C_l(x))$ denotes the first-level output obtained from a first-level classifier C_l, $l = 1, \ldots, L$. The simplest aggregation rule is to take the average of the outputs with the weight ω:

$$\mathcal{F}_1(C|\omega) \quad = \quad \frac{\sum_{l=1}^{L} \omega_{l,t} \mu(C_l(x))}{L},$$

where $\omega_{l,t}$ is a weight for each base classifier determined in the training phase. A second aggregation rule is

$$\mathcal{F}_2(C) \quad = \quad \beta_t^T R, \text{ where } \beta_t = (R_t^T R_t)^{-1} R_t^T y_t.$$

Here, $R = (1, \mu(C_1(x)), \ldots, \mu(C_L(x)))$ and $R_t = (1, \mu(C_{1,t}(x_t)), \ldots, \mu(C_{1,t}(x_t)))$ denote a collection $\{\mu(C_l(x))\}_{l=1}^{L}$ for the test set and the training set, respectively. x_ts and y_ts are the covariates and known class labels for the training set. Then, the final decision rule is

$$\hat{C} = \begin{cases} -1, & \text{if } \mathcal{F}(C) < c^*, \\ +1, & \text{if } \mathcal{F}(C) \geq c^*, \end{cases}$$

where $\mathcal{F}(C)$ can be either $\mathcal{F}_1(C|\omega)$ or $\mathcal{F}_2(C)$.

2. Aggregation by stacked generalization. Second, we employed a special type of stacked generalization which is slightly different from the procedure proposed by Wolpert [1992]. Instead of using cross-validation, we split the data into a training set (X_t, Y_t), validation set (X_v, Y_v), and testing set (X_s, Y_s). Let $C_l^0, l = 1 \ldots L$ denote level-0 classifiers, where L is the number of base classifiers, and C^1 denote a level-1 classifier. In the stage-0, training set (X_t, Y_t) is used as an input to train for the

classification task, and L learning rules are obtained. Next, we apply each learning rule to the validation set and obtain L sets of level-0 outputs $\{\mu(C_l^0(X_v))\}_{l=1}^L$. Now $\{\mu(C_l^0(X_v)), Y_v\}_{l=1}^L$ are used as level-1 inputs for the stage-1 learning algorithm to learn stage-1 aggregation rule $C^1 : \{\mu(C_l^0(X))\}_{l=1}^L \in \{-1,1\}^L \mapsto \{-1,1\}$. Stacked generalization is used to combine L sets of binary outputs $\{-1,1\}$ obtained from the level-1 classifiers.

Classification inducer: C-BSVM, AdaBoost(AdaBoost.M1), Random forests.

In this study, we employed three different learning algorithms as classification inducers: Support Vector Machine (SVM), AdaBoost (tree), and Random forests. In this section, we provide a brief review of these three learning algorithms.

1. Support Vector Machine. Support Vector Machines [Vapnik, 1998] are among the most popular machine learning algorithms based on the kernel method. SVMs exhibit state-of the art performance in the classification task in various biomedical data settings (Zhao et al. [2006]; Lienemann et al. [2007]; Thurston et al. [2009]; Yu et al. [2010]). In the classification problem, SVMs find a decision function f for a given set of attributes x, and predict the class label b of target y according to the sign of $f(x)$ as follows:

$$b(x) = sign(f(x)) = \begin{cases} +1, & \text{if } f(x) \geq 0, \\ -1, & \text{if } f(x) < 0. \end{cases}$$

SVMs provide multiple types of outputs, including a decision value $f(x) \in \mathbb{R}^1$ and a class label $b(x) \in \{-1,1\}$. Many different types of SVMs have been developed, and we utilize a bound constraint version of the C classification (C-BSVM) algorithm as a base classifier. To implement the C-BSVM algorithm, the **ksvm** function in

the **libsvm** library [Chang and Lin, 2001] in the **R** package [R Development Core Team, 2010] is utilized. In C-BSVM, the successive overrelaxation (SOR) algortim for quadratic programs is used to train SVMs by the modified **TRON** QP solver (Lin et al. [1999]; Karatzoglou et al. [2006]). For more details concerning the C-BSVM algorithm, we refer the reader to Mangasarian and Musicant [1999]. We use linear and radial basis kernels for all SVM models in data analysis, and the quadratic polynomial kernel was added for simulation studies:

$$\begin{aligned} \text{Linear kernel} &: k(x, x') := < x, x' > \\ \text{Quadratic polynomial kernel} &: k(x, x') = (\text{scale} \cdot < x, x' > + \text{offset})^2 \\ \text{Radial basis function kernel (RBF)} &: k(x, x') := \exp(-\sigma \|x - x'\|^2), \end{aligned}$$

where $< \cdot, \cdot >$ denotes inner product of two vectors, and k is a kernel function. Most internal parameters of SVM learning are obtained by the internal 5-fold CV. For the regularization margin in the Lagrange formulation, we use a default setup of 1 for a relatively simple but robust prediction function.

2. AdaBoost (AdaBoost.M1). AdaBoost (*Adaptive Boosting*; [Freund and Schapire, 1997]) generates a set of classifiers sequentially and then aggregates them by a weighted majority voting method. In the first step of AdaBoost, n observations in the training set have a weight equal to $1/n$, and at each step, the procedure updates the weight according to the classification performance in the previous step. It is interesting to note that AdaBoost improves performance by increasing weights for the examples on which the prediction is poor in the pervious step. Instead of using majority voting (i.e. voting for the most popular class), Adaboost aggregates

outcomes from the base classifiers by summing their probabilistic predictions, and then selecting the best prediction performance (weighted majority voting). Freund and Schapire [1997] proved that the VC-dimension of the final hypothesis $h_t : X \mapsto Y$ generated by AdaBoost has a finite upper bound, thus these hypotheses belong to a VC-class. In this study, we implement AdaBoost by using the **R** package **adabag**.

Algorithm 2. *AdaBoost (AdaBoost.M1) [Freund and Schapire, 1997] Assign initial weights* $\omega_i^1 = 1/n$*, where n is the number of observations in the training set. For t=1 to T,*

1. *Set* $p^t = \frac{w^t}{\sum_{i=1}^n \omega_i^t}$.
2. *Call* ***WeakLearn*** *(e.g. decision tree), providing it with the distribution* p^t*: get back a hypothesis* $h_t : X \mapsto Y$.
3. *Calculate the error of* $h_t : \epsilon_t = \sum_{i=1}^n p_i^t [h_t(x_i) \neq y_i]$*, where for any predicate* π*,* $[\pi]$ *is defined to be 1 if* π *holds and 0 otherwise. If* $\epsilon_t > 1/2$*, then set T=t-1 and abort loop.*
4. *Set* $\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$.
5. *Update weight by* $\omega_i^{t+1} = \omega_i^t \beta_t^{1-[h_t(x_i) \neq y_i]}$
6. *Repeat Step 1 to Step 5 and obtain the final output* $h_f(x) = \arg\max_{y \epsilon Y} \sum_{t=1}^T \left(\log \frac{1}{\beta_t}\right) [h_t(x_i) \neq y_i]$.

3. Random forests. As mentioned in the literature review, random forests [Breiman, 2001] are one of the most popular decision committee methods that increase diversity among base classifiers by using bootstrap samples and a random selection of features. A large number of trees are then combined by majority voting

without pruning for the individual trees. Breiman [2001] proved that the generalization error for forests converges almost surely to a limit as the number of trees increases by the Strong Law of Large Numbers. Also, the author provided an upper bound for the generalization error which is $\bar{\rho}(1-s^2)/s^2$, where $\bar{\rho}$ is the mean value of the correlation between individual trees and s is the strength of the set of trees. This indicates that random forests can improve classification performance by reducing the correlation between individual trees and improving each individual tree's performance [Zhu, 2008]. Based on experimental results using 20 datasets, Brieman noted that random forests give results competitive with AdaBoost, but are more robust with respect to noise and are computationally faster than AdaBoost. It is interesting to note that random forests can show different behaviors in runs on the data sets compared to runs on larger data sets. In the runs on larger data sets, the strength of individual trees increased while correlation between trees converged more quickly. In this study, we implemented a random forests algorithm by using the **R** package **randomForests** [Liaw and Wiener, 2002].

Algorithm 3. *Random forests [Breiman, 2001]*

1. *For each k=1 to K, draw a bootstrap sample of the training data, say y, and generate a random vector* Θ_k.
2. *Grow a maximal-depth tree using y and* Θ_k*, resulting in a classifier* $h(y, \Theta_k)$.
3. *After growing K trees, vote for the most popular class to make a final decision* $h_f(x)$.

2.3 Chemical toxicity data analysis

As mentioned in the introduction, prediction of the potential risk of chemicals is an important issue in toxicology. The quantitative structure-activity relationship

(QSAR) model is one of the most common methods used to predict chemical toxicity activity. The QSAR model focuses on studying a mathematical relationship between chemical structure and biological activity and predicting the biological activity for given descriptors of chemical structure. Therefore, prediction accuracy is one of the most important issues in the QSAR model in addition to quality of data. Due to the high-dimensionality of the descriptors, many QSAR models have been developed using learning techniques from machine learning such as support vector machines (SVMs) [Mazzatorta et al., 2006], random forests [Polishchuk et al., 2009], or neural networks (NNs) [Cherkasov, 2005], as well as statistical approaches such as partial least squares.

However, traditional QSAR models have a problem of low external prediction accuracy, and researchers have studied several approaches to overcome this problem. Zhu et al. [2008] developed various QSAR models to predict aquatic toxicity for a large set of chemical compounds. In this study, four consensus QSAR models were constructed by integrating all validated individual models in four different ways. Individual models include the kNN algorithm, SVM, multiple linear regression (MLR) with forward and backward stepwise variable selection, MLR with a genetic algorithm for variable selection, partial least squares regression (PLR) with a jackknife method to identify significant descriptors, associative neural networks (ASNN), and artificial neural networks (ANN). All consensus QSAR models outperformed any individual models with higher prediction accuracy to external validation sets. Martin et al. [2008] developed a QSAR methodology based on hierarchal clustering to predict toxicological endpoints. First, the training data were divided by the use of a variation of Ward's minimum variance clustering method based on molecular descriptor values. Next, each cluster was evaluated to see if an acceptable QSAR model could be

developed by using a genetic algorithm and a leave-one-out cross validation. Finally, the predicted toxicity for a test chemical was constructed by the weighted average of the valid predictions. This approach is useful when a chemical compound does not fall into a single chemical cluster. Zhu et al. [2009] developed a combinatorial QSAR approach including kNN, random forest, hierarchical clustering, NN, and Food and Drug Administration (FDA) MDL-QSAR model in a similar way as in Zhu et al. [2008].

Chapter 3

The change-line classification and regression method

In this chapter, we present the method and the estimating procedure for the change-line classification model and the change-line regression models. Two sets of simulation studies are provided to examine feasibility of the proposed methods. Additional experiments are conducted to study the performance of the proposed method under several situations through additional Monte-Carlo simulations. The change-line classification method and PPR are applied to the chemical toxicity data set. A preliminary study for the hypothesis test for the presence of a change-line are presented at the end of this chapter.

3.1 Model and method

3.1.1 Data set-up and assumptions

Let us suppose we observe n independent and identically distributed (i.i.d.) realizations of the random triple (Y, X, Z) in a probability space $(\mathcal{X}, \mathcal{A}, P)$ such that

$$Y(\theta; X, Z) \sim C(X; \omega, \gamma)F(Z; \beta, \tau) + \{1 - C(X; \omega, \gamma)\}G(Z; \delta, \tau), \tag{3.1}$$

where $C(X;\omega,\gamma) = \mathbf{1}\{\omega^T X - \gamma > 0\}$, $\omega = (\omega_1,\omega_2) \in \mathcal{S}^2 = \{(\omega_1,\omega_2) \in \mathbb{R}^2 : \|\omega\| = 1\}$, and $\gamma \in [a,b] \in \mathbb{R}$. We assume that $X \in \mathbb{R}^2$ is independent of random error ϵ, which satisfies $P\epsilon = 0$, $\sigma^2 = P\epsilon^2 < \infty$, and γ is a constant within a bounded interval $[a,b]$, $-\infty < a < b < \infty$. We also assume that $P\|Z\|^2 < \infty$, $Z \in \mathbb{R}^q$, and Z is independent of ϵ. Here P denotes the true probability measure.

As a notation for the parameters, let $\theta = (\zeta,\varphi)$, where $\varphi = (\omega,\gamma)$, and $\zeta = (\beta_1,\ldots,\beta_p,\delta_1,\ldots,\delta_p)$, and let $\hat{\theta}_n$ be an M-estimator, where $\hat{\theta}_n \equiv (\hat{\zeta},\hat{\varphi})$, $\hat{\varphi} = (\hat{\omega},\hat{\gamma})$, and $\hat{\zeta} = (\hat{\beta}_1,\ldots,\hat{\beta}_p,\ \hat{\delta}_1,\ldots,\hat{\delta}_p)$. Further, we assume that the true values of the model parameters from the two subgroups are different, that is, $\beta_0 \neq \delta_0$ and a change occurs at (ω_0,γ_0) for identifiability of the true parameter values θ_0. This assumption is made by many studies of change-point analysis, for examples, please see Pons [2003], Kosorok and Song [2007], and Lee and Seo [2008]. X is assumed to have a strictly bounded and positive density over $[a,b]$ with $P(\omega_0^T X < a) > 0$ and $P(\omega_0^T X > b) > 0$, where ω_0 is a true parameter. We need an additional assumption that there exists an open set $A \in \mathbb{R}^2$ such that the density of X on the closure $\bar{A}$ is bounded below and $\omega_0^T x = \gamma_0$ for some $x \in A$ (assumption **A**). Also, the density of $\omega_0^T X - \gamma$ is assumed to be positive in a neighbor of A (assumption **B**).

Since we assume that $\|\omega\| = 1$ and $\gamma \in [a,b]$, we know that ω and γ are bounded and hence $\hat{\omega}$ and $\hat{\gamma}$ exist. As Kosorok [2008b] discussed, however, this setting does not guarantee the existence of the estimator for the model parameters. In this study, we assume that the model parameter ζ ranges over some known compact set H_1 in $\mathbb{R}^p$ so that ζ is bounded, and hence $\hat{\zeta}$ exists.

Let the parameter space $H = H_1^2 \times [a, b] \times \mathcal{S}^2$. In our present setting, we have assumed that the classifier C is completely determined by the two-dimensional feature space variable X. X can be either a subset of Z or a totally different variable from Z. The primary goal of this study is to estimate θ through the least squares method or the maximum likelihood method so that we can find a line $\omega^T X - \gamma$ that divides the population into two subgroups and also obtain better estimates for the model parameters.

For the specific, change-line classification model we implement here, we assume there exist two subgroups, each of which follows a normal distribution with mean and variance (μ_0, σ_0^2) and (μ_1, σ_1^2), respectively. The corresponding model can be presented as

$$P(Y \le u|X;\theta) = C(X;\omega,\gamma)\Phi\left(\frac{u-\mu_1}{\sigma_1}\right) + \{1 - C(X;\omega,\gamma)\}\Phi\left(\frac{u-\mu_0}{\sigma_0}\right), \tag{3.2}$$

where $\theta = (\zeta, \varphi)$, $\varphi = (\omega, \gamma)$, and $\zeta = (\mu_0, \mu_1, \sigma_0^2, \sigma_1^2)$. We assume that the true value of the parameters of $(\mu_0^0, \sigma_0^{2,0})$ are not equal to $(\mu_1^0, \sigma_1^{2,0})$ for the heterogeneous variance model. Then, the loglikelihood function for model 3.2 can be written as

$$M_n(\theta) = -\frac{n_1(\theta)}{2}\{1 + \log\sigma_1^2(\theta)\} - \frac{n_0(\theta)}{2}\{1 + \log\sigma_0^2(\theta)\},$$

where $c_i(\theta) = \mathbf{1}\{\omega^T x_i - \gamma > 0\}$ for $x_i \in \mathbb{R}^2$, $i = 1, \ldots, n$, $n_0(\theta) = n - n_1(\theta)$, $n_1(\theta) = \sum_{i=1}^{n} c_i(\theta)$, $\mu_0(\theta) = \frac{\sum_{i=1}^n (1 - c_i(\theta))y_i}{n_0(\theta)}$, $\mu_1(\theta) = \frac{\sum_{i=1}^n c_i(\theta) y_i}{n_1(\theta)}$, $\sigma_0^2(\theta) = \frac{\sum_{i=1}^n (1 - c_i(\theta))(y_i - \mu_0(\theta))^2}{n_0(\theta)}$ and $\sigma_1^2(\theta) = \frac{\sum_{i=1}^n c_i(\theta)(y_i - \mu_1(\theta))^2}{n_1(\theta)}$. Thus, the change-line classification problem is to find a M-estimator $\hat{\theta}_n$ that maximizes $M_n(\theta)$ over the parameter space H, where $\hat{\theta}_n \equiv (\hat{\zeta}, \hat{\varphi})$, $\hat{\varphi} = (\hat{\omega}, \hat{\gamma})$, and $\hat{\zeta} = (\hat{\mu}_0, \hat{\mu}_1, \hat{\sigma}_0^2, \hat{\sigma}_1^2)$.

For change-line regression, in contrast, we assume that two subgroups have regression models with different regression parameters β and δ, respectively, where $\beta^0 \neq \delta^0$, such that

$$E(Y;\theta) = C(X;\omega,\gamma)\beta^T Z + \{1 - C(X;\omega,\gamma)\}\delta^T Z, \tag{3.3}$$

where $C(X) = \mathbf{1}\{\omega^T X - \gamma > 0\}$, $\theta = (\zeta, \varphi)$, $\varphi = (\omega, \gamma)$, and $\zeta = (\beta_1, \ldots, \beta_p, \delta_1, \ldots, \delta_p)$. In change-line regression, we estimate θ through least squares, which is the same as finding the M-estimator that maximizes $M_n(\theta) = n\mathbb{P}_n m_\theta$ where

$$M_n(\theta) = -\sum_{i=1}^{n} \left[y_i - C(x_i;\omega,\gamma)\beta^T z_i - \{1 - C(x_i;\omega,\gamma)\}\delta^T z_i \right]^2.$$

As before, let $\hat{\theta}_n$ be the maximizer of $M_n(\theta)$, where $\hat{\theta}_n \equiv (\hat{\zeta}, \hat{\varphi})$, $\hat{\varphi} = (\hat{\omega}, \hat{\gamma})$, and $\hat{\zeta} = (\hat{\beta}_1, \ldots, \hat{\beta}_p,\ \hat{\delta}_1, \ldots, \hat{\delta}_p)$.

3.1.2 Estimating procedure

In the present setting, the estimates are obtained in a similar way as done in Kosorok and Song [2007]. Before we begin estimation, we need to construct a set $\mathcal{S}(\omega,\gamma) = \{(\omega_1,\gamma_1), \ldots, (\omega_k,\gamma_k),\ k \leq n(n-1)\}$, consisting of all possible lines generated by two points (x_i, x_j) in $\mathbb{R}^2$ from n observations. In this thesis, $\mathcal{S}(\omega,\gamma)$ is obtained by use of the new algorithm developed by Kosorok [2008a], and described in Chapter 2.1.6. We search for γ by considering as candidates only elements of the set $\{\gamma_1, \ldots, \gamma_{n-1}\}$ consisting of the midpoints of the sorted values of $\omega^T x_1, \ldots, \omega^T x_n$.

The M-estimator $\hat{\theta}_n$ can be obtained in two steps. In the first step, for each fixed (ω,γ) in $\mathcal{S}(\omega,\gamma)$, we maximize the objective function M_n over model parameters ζ to obtain the profile objective function $pM_n(\omega,\gamma) \equiv \sup_{\zeta} M_n(\omega,\gamma,\zeta)$. In the second step, $(\hat{\omega}_n, \hat{\gamma}_n)$ can be obtained by searching for $\arg\max_{\mathcal{S}(\omega,\gamma)} pM_n(\omega,\gamma)$ using a line

search on the unit-circle for direction vector ω and on the line for a cut point γ. Let $\hat{\zeta}_n = \arg\max_{\zeta} M_n(\zeta, \hat{\omega}_n, \hat{\gamma}_n)$. By this procedure, we can obtain $\hat{\theta}_n = (\hat{\zeta}, \hat{\gamma})$ that maximizes the objective function. Estimates for the model parameter obtained in this way are $\sqrt{n}$–consistent, and the estimate for the change-point parameters are n–consistent. Also, this is an adaptive estimation procedure in the sense that we can have $\sqrt{n}$–consistent estimates for the model parameters whether or not we know the true value of the change-point parameter. We will show this in Chapter 4. Please see Pons [2003] and Kosorok and Song [2007] for similar results for the change-point model.

3.2 Simulation study

3.2.1 Change-line methods for heterogeneous subgroups

Two sets of simulation studies based on the Monte Carlo (MC) method were conducted to illustrate the applicability of the change-line classification and regression method.

Change-line classification model

The experimental data for the change-line classification model (3.2) were generated from the bivariate normal distribution with mean vector $(\mu_0, \mu_1)^T$ and covariance matrix $\Sigma = diag(\sigma_0^2, \sigma_1^2)$. For all these simulations, the true values of the parameters were chosen as $(\omega_1^0, \omega_2^0) = (-1/\sqrt{2},\ 1/\sqrt{2})$, $\gamma^0 = 0$, $(\mu_0^0, \sigma_0^{2,0}) = (2, 4)$, $(\mu_1^0, \sigma_1^{2,0}) = (0, 1)$, and $n_0 = n_1 = n/2$. For a given sample size, 100 independently replicated samples were generated from the true model. Figure 3.1 and Table 3.1 display the results of simulations based on 100 replicated samples, with sample size ranging from 50 to 600. MC mean, MC standard error of estimates (MCSE), and 95% empirical

percentiles were calculated based on the 100 runs of simulations for each sample size. MC mean was calculated by $\bar{\hat{\theta}}_n = \sum_{j=1}^{100} \hat{\theta}_n^{(j)}/100$, and MCSE was calculated by $\sqrt{\frac{\sum_{j=1}^{100}(\hat{\theta}_n^{(j)} - (\bar{\hat{\theta}}_n))^2}{100 \times (100-1)}}$, where $\hat{\theta}_n^{(j)}$ denotes the estimate of the parameter θ from the j^{th} set of data, $j = 1, \ldots, 100$. The MC radius mean for (ω_1, ω_2) was calculated as $(\cos \bar{\hat{u}}, \sin \bar{\hat{u}})$, where $\hat{u}$ is estimate of the angle between the line $\hat{\omega}^T X$ and the X-axis. Our method worked very well even for small sample sizes, and the MC mean of the estimates were close to the true values with smaller MCSE as sample size increased for both model parameters and change-line parameters.

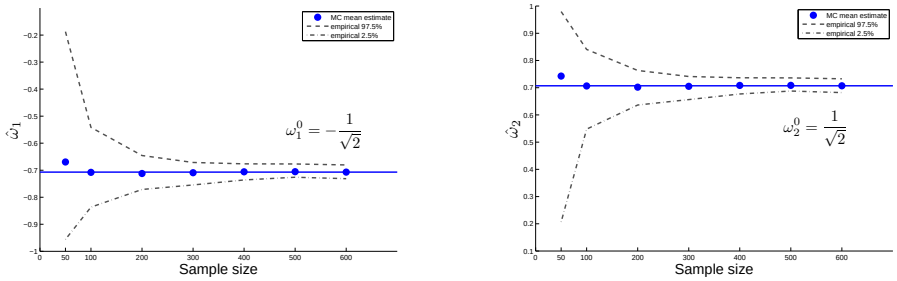
To check the existence of an underlying change in the distribution of Y, a graphic examination was performed by displaying the Gaussian kernel estimates of the mean and standard deviation of Y. The Gaussian kernel estimates for the mean and standard deviation were calculated by

$$\begin{aligned} \hat{\mu}(y|u) &= \frac{\frac{1}{nh}\sum_{i=1}^{n} K(\frac{u-u_i}{h}) y_i}{\frac{1}{nh}\sum_{i=1}^{n} K(\frac{u-u_i}{h})} \\ \hat{\sigma}(y|u) &= \sqrt{\frac{\frac{1}{nh}\sum_{i=1}^{n} K(\frac{u-u_i}{h})(y_i - \hat{\mu}(y|u))^2}{\frac{1}{nh}\sum_{i=1}^{n} K(\frac{u-u_i}{h})}}, \end{aligned}$$

where $u_i = \hat{\omega}^T x_i, i = 1, \ldots, n = 600$, K denotes the Gaussian kernel function, and h is the bandwidth. We used the MC radius mean from the sample of size 600 for $\hat{\omega}$. Then we regressed the Gaussian kernel mean and standard deviation above on u by Robust Locally Weighted Least Squares Scatterplot Smoothing and a 1^{st} degree polynomial model (RLOWESS) from the **MATLAB** package (see Cleveland [1979]). Figure 3.2 shows the RLOWESS regression line of the Gaussian kernel estimates for the mean and standard deviation. The bandwidth for the Gaussian kernel was 0.2487, which was calculated by Silverman's optimal bandwidth calculation for the normal kernel

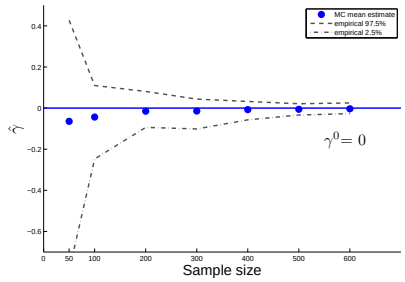
Table 3.1: Summary statistics from 100 replications of simulation study for the change-line classification under various sample sizes : $(50, 100, 200, 300, 400, 500, 600)$. For all simulations, true values of the parameter were chosen as $(\omega_1^0 = -1/\sqrt{2}, \omega_2^0 = 1/\sqrt{2}, \gamma^0 = 0, \mu_0^0 = 2, \mu_1^0 = 0, \sigma_0^{2,0} = 4, \sigma_1^{2,0} = 1)$.

		Sample Size for each iteration						
Parameter		50	100	200	300	400	500	600
α = **2.356**	MC Mean	2.304	2.357	2.363	2.359	2.355	2.354	2.356
	MCSE	0.033	0.010	0.004	0.003	0.002	0.002	0.002
ω_1 = **−0.707**	MC Radius Mean	-0.670	-0.708	-0.712	-0.709	-0.706	-0.706	-0.707
	MCSE	0.024	0.007	0.003	0.002	0.001	0.001	0.001
ω_2 = **0.707**	MC Radius Mean	0.743	0.706	0.702	0.705	0.708	0.709	0.707
	MCSE	0.018	0.007	0.003	0.002	0.001	0.001	0.001
γ = **0**	MC Mean	-0.064	-0.044	-0.015	-0.014	-0.008	-0.006	-0.004
	MCSE	0.028	0.011	0.005	0.003	0.002	0.001	0.001
μ_0 = **2**	MC Mean	2.099	2.053	2.026	2.025	2.014	2.023	2.028
	MCSE	0.049	0.030	0.017	0.016	0.014	0.012	0.011
μ_1 = **0**	MC Mean	-0.029	-0.007	-0.017	-0.012	-0.011	-0.010	-0.010
	MCSE	0.036	0.016	0.010	0.007	0.006	0.006	0.005
σ_0^2 = **4**	MC Mean	3.693	3.851	3.855	3.871	3.919	3.927	3.950
	MCSE	0.140	0.098	0.062	0.046	0.038	0.034	0.031
σ_1^2 = **1**	MC Mean	0.931	0.970	0.972	0.984	0.986	0.985	0.987
	MCSE	0.041	0.024	0.014	0.011	0.010	0.009	0.008

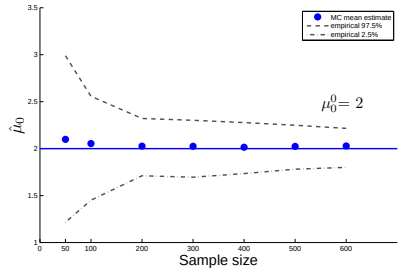


(a) ω_1

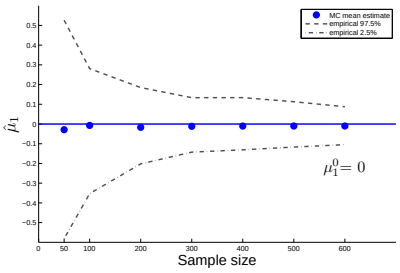
(b) ω_2



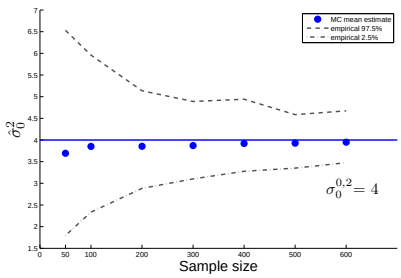
(c) γ



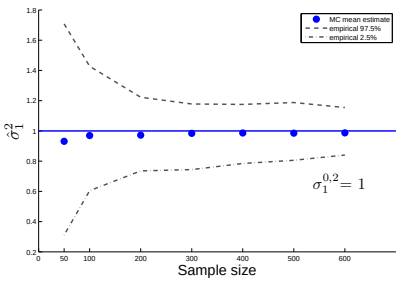
(d) μ_0



(e) μ_1

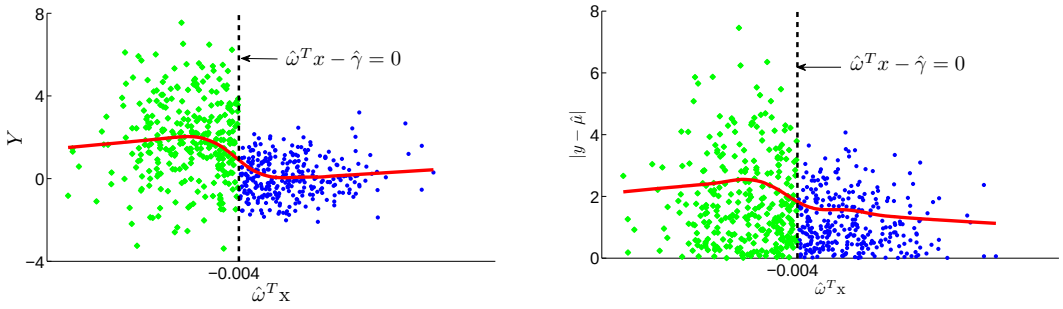


(f) σ_0^2



(g) σ_1^2

Figure 3.1: Plots of MC mean of the estimate based on 100 replications per scenario for the change-line classification under various sample sizes: 50, 100, 200, 300, 400, 500, 600. For all simulations shown, the true values of the parameter were chosen as $\omega_1^0 = -1/\sqrt{2}, \omega_2^0 = 1/\sqrt{2}, \gamma^0 = 0, \mu_0^0 = 2, \mu_1^0 = 0, \sigma_0^{2,0} = 4, \sigma_1^{2,0} = 1$. Dots denote MC mean estimates, dashed lines denote 95% empirical percentiles of the estimates, and solid lines are the true value of the parameters.



(a) Change in mean

(b) Change in standard deviation

Figure 3.2: Robust Locally Weighted Least Squares Scatterplot Smoothing (RLOWESS) curves for the Gaussian Kernel mean and standard deviation for the simulated data of size 600. $\hat{\omega}$ and $\hat{\gamma}$ were obtained by the change-line classification method. Dots denotes observed y and residual $\sqrt{(y - \hat{\mu}(y)_{\text{Gaussian}})^2}$ for mean and standard deviation, respectively. Solid lines denote RLOWESS regression lines for the Gaussian kernel estimates with bandwidth of 0.2487.

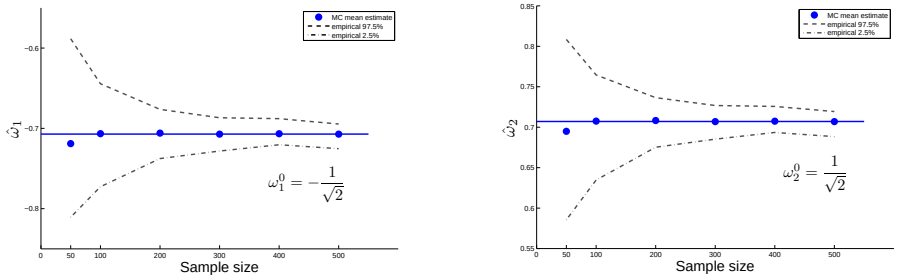
where $h \approx 0.9 \times A \times n^{-1/5}$, and where $A = \min(sd(u), IQR(u)/1.34)$ [Silverman, 1986]. For both mean and standard deviation, a change occurs at near zero, which is the true value of the cut-point γ^0.

Change-line regression model

Next, simulation data for the change-line regression model (3.3) were generated with the true values of the parameters $(\omega_1^0, \omega_2^0) = (-1/\sqrt{2}, 1/\sqrt{2})$, $\gamma^0 = 0$, $(\beta_1^0, \beta_2^0) = (2, 1)$, $(\delta_1^0, \delta_2^0) = (-2, -1)$, and $n_0 = n_1 = n/2$. If we fit a regression model to the whole population of these data, the overall effect of X would be canceled out and we would find no regression effect. Figure 3.3 and Table 3.2 display the MC mean, MCSE, and 95% empirical percentiles based on 100 replications for each scenario for the change-line regression for sample sizes ranging from 50 to 500. The estimates of both model parameters and change-line parameters show a good approximation to the true parameters. The precision of all estimates increased as sample size increased.

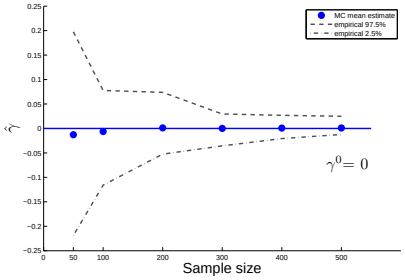
Table 3.2: Summary statistics from 100 replications of the simulation study for change-line regression under various sample sizes : $(50, 100, 200, 300, 400, 500)$. For all simulations, the true values of the parameter were chosen as $(\omega_1^0 = -1/\sqrt{2}, \omega_2^0 = 1/\sqrt{2}, \gamma^0 = 0, \beta_1^0 = 2, \beta_2^0 = 1, \delta_1^0 = -2, \delta_2^0 = -1)$.

Parameter		Sample Size for each iteration					
		50	100	200	300	400	500
α = **2.356**	Mean	2.3731	2.3555	2.3545	2.3564	2.3556	2.3561
	MCSE	0.0091	0.0032	0.0022	0.0014	0.0011	0.0010
ω_1 = **−0.707**	MC Radius Mean	-0.7190	-0.7066	-0.7059	-0.7072	-0.7067	-0.7070
	MCSE	0.0062	0.0046	0.0015	0.0010	0.0008	0.0007
ω_2 = **0.707**	MC Radius Mean	0.6950	0.7076	0.7083	0.7070	0.7075	0.7072
	MCSE	0.0066	0.0032	0.0016	0.0010	0.0008	0.0007
γ = **0**	MC Mean	-0.0128	-0.0063	0.0011	0.0001	0.0007	0.0002
	MCSE	0.0105	0.0048	0.0027	0.0016	0.0012	0.0010
β_1 = **2**	MC Mean	1.9692	1.9948	1.9974	1.9989	1.9926	1.9991
	MCSE	0.0231	0.0147	0.0107	0.0079	0.0070	0.0064
β_2 = **1**	MC Mean	1.0019	1.0158	1.0078	1.0102	1.0012	1.0018
	MCSE	0.0219	0.0163	0.0120	0.0096	0.0074	0.0064
δ_1 = **−2**	MC Mean	-2.0046	-1.9967	-2.0077	-2.0048	-2.0033	-2.0038
	MCSE	0.0243	0.0158	0.0116	0.0085	0.0072	0.0059
δ_2 = **−1**	MC Mean	-1.0130	-0.9888	-0.9991	-0.9992	-0.9970	-0.9921
	MCSE	0.0230	0.0152	0.0103	0.0080	0.0071	0.0060

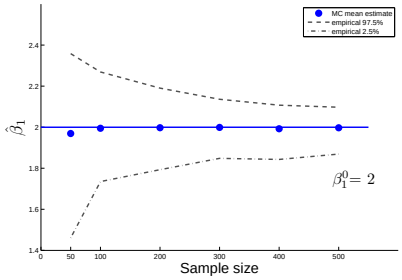


(a) ω_1

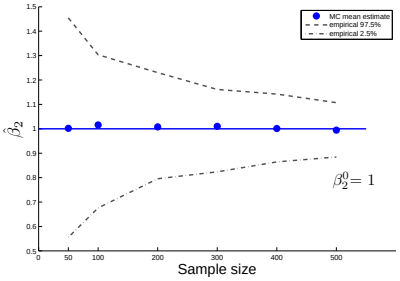
(b) ω_2



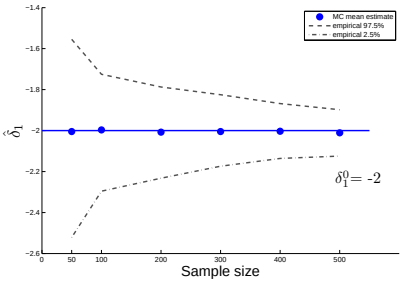
(c) γ



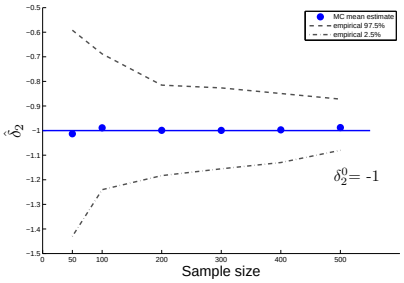
(d) β_1



(e) β_2

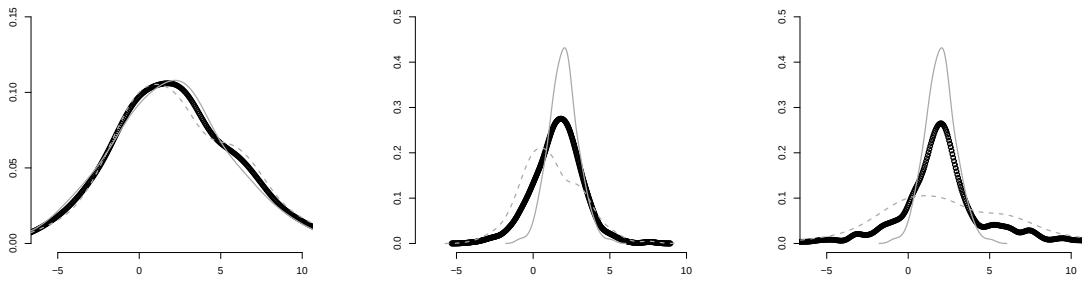


(f) δ_1



(g) δ_2

Figure 3.3: Plots of MC mean of the estimates from 100 replications per scenario for the change-line regression under various sample sizes : 50, 100, 200, 300, 400, 500. The true values of the parameters were chosen as $\omega_1^0 = -1/\sqrt{2}$, $\omega_2^0 = 1/\sqrt{2}$, $\gamma^0 = 0$, $\beta_1^0 = 2$, $\beta_2^0 = 1$, $\delta_1^0 = -2$, $\delta_2^0 = -1$. Dots denotes MC mean estimates, dashed lines denote 95% empirical percentile of the estimates, and solid lines denote the true values of the parameters.



(a) scenario 1 (b) scenario 2 (c) scenario 3

Figure 3.4: Density plot for (a): $\mathbf{N(2,4)}$ vs. $\mathbf{N(2,4)}$ (b): $\mathbf{N(1,4)}$ vs. $\mathbf{N(2,1)}$ (c): $\mathbf{N(2,4)}$ vs. $\mathbf{N(2,1)}$. Solid black curve is density plot for combined population. Solid gray curve denotes density for the group which has smaller variance and dashed gray curve denotes density for the group which has greater variance in scenarios 2 and 3. Density curves were obtained by using the Gaussian kernel method.

3.2.2 Change-line methods under various scenarios

A small investigation was carried out to see if the change-line classification model can work for less favorable cases such as the followings:

Simulation 1: $Z_0 \sim \mathbf{N}(2,4)$ vs. $Z_1 \sim \mathbf{N}(2,4)$

Simulation 2: $Z_0 \sim \mathbf{N}(1,4)$ vs. $Z_1 \sim \mathbf{N}(2,1)$

Simulation 3: $Z_0 \sim \mathbf{N}(2,4)$ vs. $Z_1 \sim \mathbf{N}(2,1)$.

Figure 3.4 displays three scenarios. In the first scenario, we considered the case when there is no heterogeneous subgroup in the completed data. In the second scenario, the subgroup lying above the change-line ($\omega^T X > \gamma$) has a greater mean but smaller variance than the other subgroup satisfying ($\omega^T X \leq \gamma$). As given in Figure 3.4(b), the two density plots do not overlap, but it is hard to tell whether or not there are two subgroups by looking at the density plot of a collection of Y. In the third scenario, two subgroups have the same mean, but one group has greater variance than the other group. Under this scenario, one group is nested in the other group, thus it

is very difficult to separate it into two subgroups having different variances.

The simulation results are given in Table 3.3. In Simulation 1, the estimated change-line parameters using the change-line method were far from the true values of the parameters. This is not a surprise because there is no underlying change-line in the simulated data. The estimated mean and variance were quite close to the true values, especially in the large sample (1.973 and 1.998 for the mean, and 4.053 and 3.219 for the variance). In Simulation 2, estimates for both change-line parameters and model parameters by the change-line method were very close to the true parameters. In Simulation 3, estimates for both change-line parameters and model parameters were close to the true parameters for the large data, and were fairly close even for the small data set.

Similarly, we ran a small experiment to investigate how the proposed change-line regression method works when there are no heterogeneous subgroups in a given population. We simulated data to satisfy $\beta_0 = \delta_0 = (2, 1)^T$ for both subgroups for sample sizes of 50 and 300. Given in Table 3.4, the estimated model parameters were quite close to the true values of parameters, especially for the large data set. Again, the estimated change-line parameters are far from the true parameters, but this is reasonable, since there is indeed no change-line in the simulated data.

These two sets of simulation studies show that the proposed change-line method can provide good estimates for model parameters even in the case where two latent subgroups are not well separated.

Table 3.3: Summary statistics from 100 replications of simulation study for the change-line classification for three different scenarios of 1, 2 and 3 under sample size of $(50, 300)$. For all simulations, true values of the change-line parameter were chosen as $(\omega_1^0 = -1/\sqrt{2}, \omega_2^0 = 1/\sqrt{2}, \gamma^0 = 0)$.

		Scenario 1		Scenario 2		Scenario 3	
		$Z_0 \sim \mathbf{N(2,4)}$ vs.$Z_1 \sim \mathbf{N(2,4)}$		$Z_0 \sim \mathbf{N(1,4)}$ vs. $Z_1 \sim \mathbf{N(2,1)}$		$Z_0 \sim \mathbf{N(2,4)}$ vs. $Z_1 \sim \mathbf{N(2,1)}$	
Parameter		50	300	50	300	50	300
$\alpha = \mathbf{2.356}$	MC Mean	1.678	1.393	2.267	2.359	2.063	2.358
	MCSE	0.082	0.092	0.053	0.007	0.072	0.007
$\omega_1 = \mathbf{-0.707}$	MC Radius Mean	-0.107	0.177	-0.641	-0.709	-0.472	-0.708
	MCSE	0.065	0.071	0.041	0.004	0.057	0.005
$\omega_2 = \mathbf{0.707}$	MC Radius Mean	0.994	0.984	0.767	0.705	0.881	0.706
	MCSE	0.042	0.048	0.028	0.005	0.034	0.0053
$\gamma = \mathbf{0}$	MC Mean	-0.205	0.178	-0.050	-0.013	-0.036	-0.022
	MCSE	0.111	0.115	0.052	0.005	0.068	0.007
μ_0	MC Mean	1.742	1.973	0.786	0.985	1.901	2.008
	MCSE	0.121	0.061	0.078	0.016	0.086	0.016
μ_1	MC Mean	2.096	1.998	1.987	1.993	1.982	1.986
	MCSE	0.095	0.058	0.039	0.007	0.041	0.008
σ_0^2	MC Mean	2.677	4.053	3.579	3.930	3.778	3.984
	MCSE	0.364	0.120	0.175	0.046	0.355	0.047
σ_1^2	MC Mean	2.766	3.218	0.971	0.981	0.902	0.972
	MCSE	0.208	0.145	0.093	0.012	0.074	0.011

Table 3.4: Summary statistics from 100 replications of simulation study for the change-line regression when there were no heterogeneous subgroups. For each simulation scenario, the sample sizes were $(50, 300)$. For all simulations, true values of the change-line parameter were chosen as $(\omega_1^0 = -1/\sqrt{2}, \omega_2^0 = 1/\sqrt{2}, \gamma^0 = 0)$.

Parameter		Sample Size for each iteration	
		50	300
$\alpha = \mathbf{2.356}$	Mean	1.409	1.446
	MCSE	0.096	0.091
$\omega_1 = \mathbf{-0.707}$	MC Radius Mean	0.161	0.124
	MCSE	0.074	0.070
$\omega_2 = \mathbf{0.707}$	MC Radius Mean	0.987	0.992
	MCSE	0.052	0.032
$\gamma = \mathbf{0}$	MC Mean	-0.019	0.198
	MCSE	0.102	0.143
$\beta_1 = \mathbf{2}$	MC Mean	1.796	1.985
	MCSE	0.105	0.051
$\beta_2 = \mathbf{1}$	MC Mean	1.043	0.986
	MCSE	0.072	0.086
$\delta_1 = \mathbf{2}$	MC Mean	2.191	1.985
	MCSE	0.138	0.042
$\delta_2 = \mathbf{1}$	MC Mean	1.066	0.964
	MCSE	0.117	0.047

3.3 Example: Chemical toxicity

To provide a practical demonstration of the proposed method, change-line classification was applied to a subset of the chemical toxicity data described in Zhu et al. [2009]. The acute toxicity data of organic chemical compounds in the rat caused by oral exposure to chemicals were utilized. The data consist of 5917 chemical compounds with toxicity activity Y and more than 800 chemical descriptors X, originally collected from different sources (National Library of Medicine [database, 2008]). The acute toxicity activity presents the median lethal dose of a toxic substance in the negative log scale ($-\log LD_{50}$). LD_{50} is the dose level required to kill 50% of the animals of a tested population. The chemical descriptors came from the following nine different classes: 2D-autocorrelations (calculated from the topological and atomic mass), 1D-functional group counts, 2D-eigenvalue-based indices (calculated by eigenvalues of the Burden matrix), 2D-molecular property (measure of certain physical properties), 2D-atom-centered fragment count, 2D-topological descriptors (number of topological pattern), 2D-connectivity indices (number of index), 0D-constitutional descriptors (number of atoms), and 2D-walk and path counts.

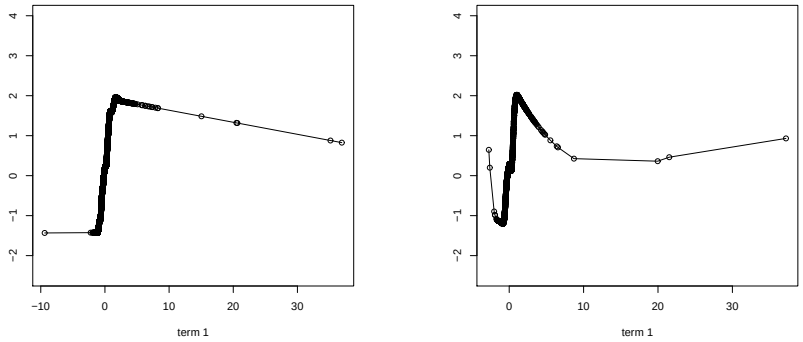
First, to reduce the dimension of the descriptors, we conducted Shrinkage principal component analysis (SPCA) for each of the nine classes of descriptors separately. SPCA is a modified PCA method recently developed by Zou et al. [2009] to automatically adjust for the correlation (linkage disequilibrium) among nearby markers in order to identify true hidden substructures (ethnicity groups) in Genome-wise association scans. For toxicity data, we found that many descriptors are similar measures and highly correlated, which tends to dominate the PCA results. The SPCA weighted down those descriptors and provided better summary measures. Please see Zou et al. [2009] for more details. Then, the first two principal components (PCs) from each of

the nine groups were picked. The primary goal of this application is to separate 5917 chemicals into two subgroups by finding the best line which is derived completely by a pair of PCs from the eighteen PCs. To choose the "best" pair of PCs, we repeated the same analysis for all of the 153 possible pairs composed from any two PCs out of the eighteen possible PCs.

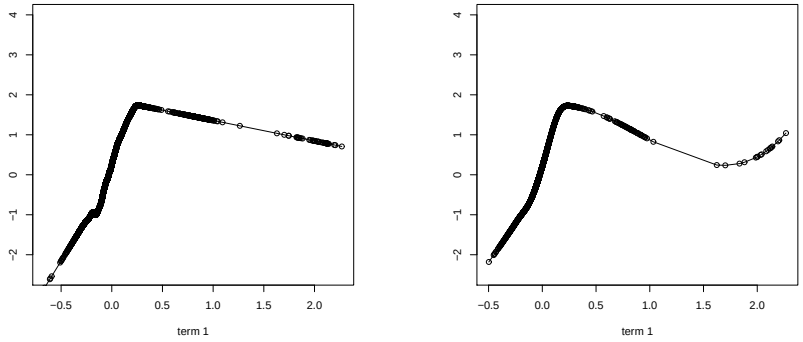
3.3.1 Projection pursuit regression

Before we proceeded with applying our proposed method, we utilized PPR to explore the data. As described in the literature review in Chapter 2.1.1, we found a sum of several linear combinations of the predictors, $f(X) = \sum_{m=1}^{M} g_m(\omega_m^T X)$. We studied the relationship between toxicity activity and the eighteen PCs first, accepting one projection (i.e., $M = 1$). Then, we repeated the same analysis with the best pair of PCs, the first PC of 2D-connectivity indices (2D connectivity index PC1) and the second PC of 2D-eigenvalues-based indices (2D eigenvalue index PC2), found by change-line classification in a subsample of size 500. We utilized Friedman's Super smoother (Supsmu) and generalized cross-validation (GCV) from the **R** package to smooth the ridge function g. For CGV, we only used 30% of the data because GCV in **R** allows only samples smaller than 2500.

Table 3.5 displays the estimates of ridge coefficient $\hat{g}_1$ and $\hat{\omega}_1$ from PPR. Based on the estimates of the projection direction $\hat{\omega}_1$, toxicity activity depends mostly on the sum of 2D-eigenvalue index PC1, 2D-eigenvalue index PC2, and 2D-connectivity index PC1 minus 2D-connectivity index PC2. When PPR was applied to only two PCs, toxicity activity depends on 2D eigenvalue index PC2 alone. Figure 3.5 shows the smoothed lines plotted against their corresponding linear combinations including all eighteen PCs and two PCs, using either Supsmu or the GCV method, respectively.



(a) Using n=5917, supsmu, 18PCs (b) Using n=1972, GCV, 18PCs



(c) Using n=5917, supsmu, 2PCs (d) Using n=1972, GCV, 2PCs

Figure 3.5: Projection pursuit regression plot: Smoothed $\hat{f} = \hat{g}_1(x;\hat{\omega})$ against the projection $(\hat{\omega}^T x)$ are plotted. Each PPR includes either all eighteen PCs (upper panel) or only two PCs (lower panel), using either Supsmu (left panel) with n=5917 or GCV (right panel) with n=1972.

The mean of $\hat{f}$ in $(-\infty, 0)$ is smaller than the mean of $\hat{f}$ in $(0, \infty)$, overall. All figures show a change in $\hat{f}$ at or near zero for $\hat{\omega}^T X$. This appears to be consistent with the result from the change-line classification method below in which the cut-point γ in the change-line classification model was estimated to be near zero.

3.3.2 Change-line classification method

Now, we describe the results of applying the change-line classification method to the chemical toxicity data. We assumed that the toxicity activity Y follows model (3.2)

Table 3.5: The estimates of ridge coefficient $\hat{g}_m$ and $\hat{\alpha}_m$ from projection pursuit regression. Supsmu and GCV were used to smooth the ridge function. 5917 chemicals were used for Supsmu, and 1972 chemicals were used for GCV.

	All 18 PCs		2 PCs	
	Supsmu	GCV	Supsmu	GCV
Ridge term	0.4690	0.4682	0.4464	0.4490
2D-autocorrelation PC1	-0.1622	-0.0793		
2D-autocorrelation PC2	0.1548	**0.3593**		
1D-functional group PC1	-0.1009	-0.1306		
1D-function group PC2	0.0251	0.0878		
2D-Eigenvalue index PC1	**0.3506**	0.1661		
2D-Eigenvalue index PC2	**0.2314**	**0.2968**	**0.9980**	**0.9981**
2D-Molecular property PC1	0.0031	0.0042		
2D-Molecular property PC2	0.0020	0.0051		
2D-atom-centered fragment PC1	0.0658	0.0822		
2D-atom-centered fragment PC2	0.0186	0.0138		
2D-topological descriptor PC1	0.0000	0.0000		
2D-topological descriptor PC2	0.0004	0.0000		
2D-connectivity index PC1	**0.4645**	**0.6428**	0.0630	0.0623
2D-connectivity index PC2	**-0.7235**	**-0.5342**		
0D-constitutional descriptor PC1	0.0445	0.0373		
0D-constitutional descriptor PC2	-0.1283	-0.1305		
2D-walk and path PC1	0.0001	0.0000		
2D-walk and path PC2	-0.0113	-0.0095		

Accepting one projection(term=1)

Super smoother span control=0

(automatic span selection by local cross validation)

Penalty for the GCV selection (gcvpen)=2

under the heterogeneous variance assumption ($\sigma_0^2 \neq \sigma_1^2$) and homogeneous assumption ($\sigma_0^2 = \sigma_1^2$), separately. $X \in \mathbb{R}^2$ is any pair of PCs produced by SPCA. To reduce computational burden, 100 independently replicated subsamples of sizes $n = 300$, 400, and 500 from the complete observations were used for each analysis. We applied the change-line classification method for all 153 combinations of the two PCs from eighteen PCs at each sample size. We restricted each subgroup to have a cluster size which is at least 5% of the total sample size. Table 3.6 summarizes the top 5 pairs of PCs from 100 replications for each sample size. The numbers in the parentheses denote ranked frequency within the top 10 pairs and the average ranking (1 for the best) based on the 100 replications. The ranking was based on the maximum value of the likelihood produced by each pair of PCs among 153 pairs. As mentioned before, the pair of 2D-connectivity index PC1 and 2D-eigenvalue based index PC2 showed the best result. This pair ranked within the top 10 most frequently (82, 87, and 96 times out of 100 replications) and showed higher rankings (7.2, 5.9, and 4.3) for the sample sizes 300, 400, and 500 compared to other pairs.

Once we decided the best pair of PCs based on the small subsample analysis, we fitted the change-line model on larger subsets of data :(300, 400, 500, 800,1000) under the heterogeneous variance assumption and (300, 400, 500) under the homogeneous variance assumption. Figure 3.6 shows the plots of the estimates for the change-line parameters and a cut-point as well as the mean and variance for the two subgroups. Under the heterogeneous variance assumption, the MC radius mean (MCSE) of $\hat{\omega}_1$ for the 2D-connectivity index PC1 was 0.00641 (0.0029) and the MC radius mean (MCSE) of $\hat{\omega}_2$ for the 2D-eigenvalue index PC2 was 0.9979 (0.0002) in subsample size of 1000. The MC mean (MCSE) of $\hat{\gamma}$ was 0.0002 (0.0043). For the homogeneous variance model, the MC radius mean (MCSE) of $\hat{\omega}_1$ was 0.0841 (0.0060), and the

MC radius mean (MCSE) of $\hat{\omega}_2$ was 0.9965 (0.0007) in subsample size of 500. The MC mean(MCSE) of $\hat{\gamma}$ was 0.0208 (0.0054). It appeared that the direction of the line primarily depended on the 2D-eigenvalue index PC2, and the cut-point was estimated near zero.

Based on the 100 replications of the subsampling with subsample size of 1000 under the heterogeneous variance model, the mean and variance of the toxicity activity were estimated to be 3.0779 and 1.0521 for the group of chemicals satisfying $\{0.0641 \times$ 2D-connectivity index PC1 $+ 0.9979 \times$ 2D-eigenvalue index PC2 $- 0.0002 > 0\}$, and 2.22 and 0.4951 for the group of chemicals satisfying $\{0.0641 \times$2D- connectivity index PC1+ $0.9979 \times$ 2D-eigenvalue index PC2 $- 0.0002 \leq 0\}$, respectively. Under the homogeneous variance model with subsample size of 500, the mean of toxicity activity was estimated to be 3.1280 and 2.2277 for the group of chemicals satisfying $\{0.0841 \times$ 2D-connectivity index PC1 $+ 0.9965 \times$ 2D-eigenvalue index PC2 $- 0.0208 > 0\}$ and for the other group of chemicals, respectively. The estimate of the common variance was 0.6920. The overall sample mean and variance of toxicity activity based on the 5917 chemical compounds were 2.5535 and 0.8801. By the line derived from the 2D connectivity index PC1 and 2D eigenvalue index PC2, 5917 chemicals can be separated into two subgroups, one group has greater mean and variance of toxicity activity compared to the other group.

In order to check the existence of an underlying change in the distribution of the toxicity activity, we conducted same graphic examination in a simulation study. $\hat{\omega}$ is the MC radius mean estimated from the subsample of size 1000, and $u_i = \hat{\omega}^T x_i, i = 1, \ldots, n = 5917$. The bandwidth for the Gaussian kernel by Silverman's formula was 0.0272 and 0.0293 under the unequal variance and the equal variance assumptions,

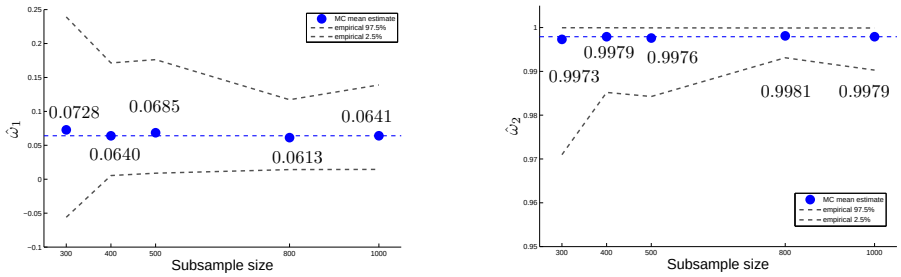
Table 3.6: Top 5 pairs of PCs from 100 replications of the change-line classification procedure for the Chemical Toxicity data under various subsample sizes, under heterogeneous variance: $(300, 400, 500)$. The numbers in the parentheses denote ranked frequency within the top 10 pairs and the average ranking (1 for best) based on the 100 replications. For example, the X14/X7 pair ranked within the top 10 for 82 times out of 100 replications, and the average ranking was 7.2 at sample size 300. The ranking was based on the maximum value of the likelihood produced by each pair of PCs among the 153 pairs.

	Subsample Size		
Rank	300	400	500
1	**X14/X7** (82 , 7.2)	**X14/X7** (87 , 5.9)	**X14/X7** (96 , 4.3)
2	X5/X7 (80 , 6.6)	X6/X7 (87 , 6.3)	X16/X7 (95 , 4.9)
3	X6/X7 (79 , 6.9)	X16/X7 (84 , 7.1)	X6/X7 (95 , 4.4)
4	X16/X7 (74 , 7.9)	X5/X7 (83 , 5.4)	X5/X7 (92 , 4.5)
5	X7/X13 (68 , 8.6)	X7/X13 (78 , 7.0)	X8/X7 (74 , 8.5)

X2: 2D-autocorrelation PC1
X3: 2D-autocorrelation PC2
X4: 1D-functional group PC1
X5: 1D-functional group PC2
X6: 2D-eigenvalue index PC1
X7: 2D-eigenvalue index PC2
X8: 2D-molecular property PC1
X9: 2D-molecular property PC2
X10: 2D-atom-centered fragment PC1
X11: 2D-atom-centered fragment PC2
X12: 2D-topological descriptor PC1
X13: 2D-topological descriptor PC2
X14: 2D-connectivity index PC1
X15: 2D-connectivity index PC2
X16: 0D-constitutional descriptor PC1
X17: 0D-constitutional descriptor PC2
X18: 2D-walk and path PC1
X19: 2D-walk and path PC2

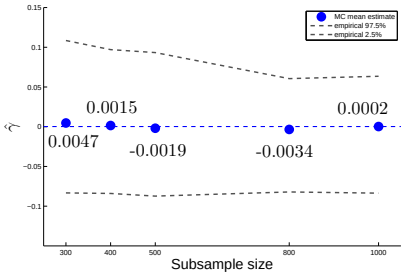
Table 3.7: Summary statistics from 100 replications of change-line classification for Chemical Toxicity data under various subsample sizes, under heterogeneous variance $(300, 400, 500, 800, 1000)$ and under homogeneous variance: $(300, 400, 500)$. Best means the set of estimators produced the maximum value of the likelihood while Worst means the set of estimators produced the minimum value of the likelihood among 100 trials.

		Subsample Size							
		Heterogeneous variance					Homogeneous variance		
Parameter		300	400	500	800	1000	300	400	500
u	Mean	**1.4979**	**1.5067**	**1.5022**	**1.5095**	**1.5066**	**1.4954**	**1.4921**	**1.4866**
	MCSE	0.0074	0.0048	0.0045	0.0030	0.0029	0.0080	0.0055	0.0061
ω_1	MCRM	**0.0728**	**0.0640**	**0.0685**	**0.0613**	**0.0641**	**0.0753**	**0.0786**	**0.0841**
	MCSE	0.0073	0.0047	0.0045	0.0030	0.0029	0.0079	0.0054	0.0060
ω_2	MCRM	**0.9973**	**0.9979**	**0.9976**	**0.9981**	**0.9979**	**0.9972**	**0.9969**	**0.9965**
	MCSE	0.0012	0.0005	0.0004	0.0003	0.0002	0.0012	0.0005	0.0007
γ	Mean	**0.0047**	**0.0015**	**-0.0019**	**-0.0034**	**0.0002**	**0.0263**	**0.0272**	**0.0208**
	MCSE	0.0057	0.0048	0.0050	0.0041	0.0043	0.0067	0.0058	0.0054
μ_0	Mean	**2.1972**	**2.2080**	**2.2031**	**2.2144**	**2.2200**	**2.2214**	**2.2342**	**2.2277**
	MCSE	0.0102	0.0096	0.0096	0.0086	0.0084	0.0113	0.0098	0.0096
μ_1	Mean	**3.1072**	**3.0906**	**3.0715**	**3.0698**	**3.0779**	**3.1709**	**3.1539**	**3.1280**
	MCSE	0.0191	0.0169	0.0157	0.0141	0.0137	0.0215	0.0179	0.0168
σ_0^2	Mean	**0.4663**	**0.4788**	**0.4759**	**0.4888**	**0.4951**			
	MCSE	0.0079	0.0068	0.0068	0.0054	0.0055			
σ_1^2	Mean	**1.0442**	**1.0422**	**1.0266**	**1.0399**	**1.0521**			
	MCSE	0.0144	0.0146	0.0127	0.0119	0.0110			
σ^2	Mean						**0.6868**	**0.6934**	**0.6920**
	MCSE						0.0057	0.0054	0.0049

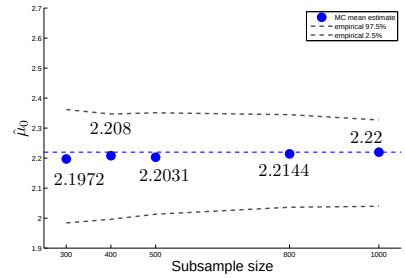


(a) ω_1

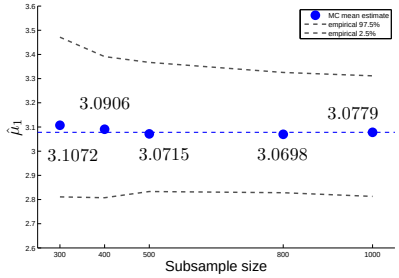
(b) ω_2



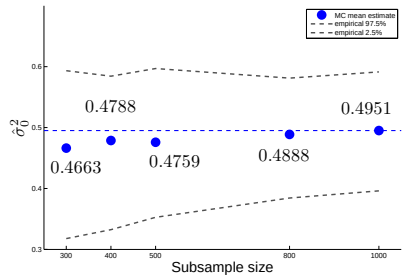
(c) γ



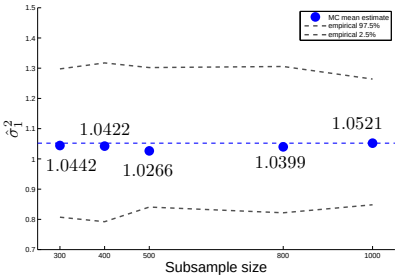
(d) μ_0



(e) μ_1



(f) σ_0^2



(g) σ_1^2

Figure 3.6: Plots of MC mean of the estimates obtained from the change-line classification of the chemical toxicity data assuming heterogeneous variance. 100 subsamples of the chemical compound were used for the analysis at each subsample sizes : $(300, 400, 500, 800, 1000)$. Dots denotes MC mean estimates, dotted lines denote the 95% empirical percentiles of estimates, and solid lines are true values of parameters.

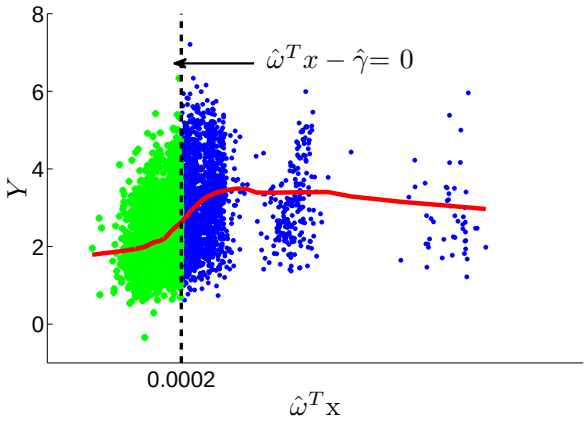
respectively. Figure 3.7 shows the RLOWESS regression lines for the Gaussian kernel mean and standard deviation estimates under the unequal variance assumption only. We can see that a change occurs near zero of $u = \hat{\omega}^T x$ for both mean and standard deviation, and this supports our finding from the change-line classification. On the other hand, these plots suggest that all chemicals can be divided into three groups according to the mean and variance of toxicity activity. In this dissertation we have restricted our research to the model in which two latent subgroups exist in a population, but we will discuss the issue of more than two groups later in the discussion.

3.4 Preliminary hypothesis test for the presence of a change-line

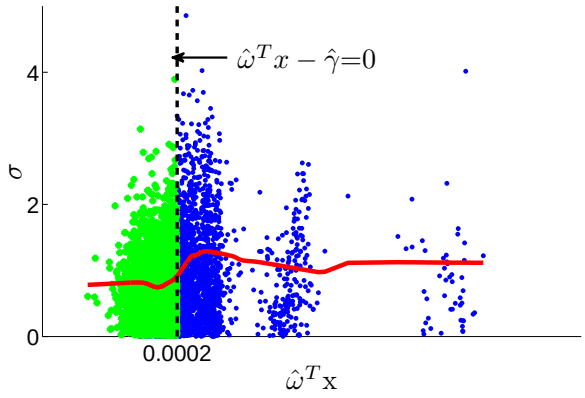
In the previous section, we observed that a graphical examination was useful to verify the existence of the underlying change-line. As a statistician, however, it is always of interest to develop a statistical hypothesis test. In this section, a small investigation was carried out to conduct a hypothesis test for the presence of a change-line, focusing on the change-line classification model. The density for a change-line classification model is written as

$$f(\theta; x, y) = C(x)\frac{1}{\sqrt{2\pi\theta_1}}\exp\left(-\frac{(y-\theta_1)^2}{2\theta_2}\right) + (1-C(x))\frac{1}{\sqrt{2\pi(\theta_1+\beta)}}\exp\left(-\frac{(y-(\theta_1+\alpha))^2}{2(\theta_2+\beta)}\right)+\varepsilon, \tag{3.4}$$

where $\theta = (\zeta, \varphi)$, $\zeta = (\alpha, \beta, \theta_1, \theta_2)(\beta > 0, \theta_2 > 0)$, $\varphi = (\omega_1, \omega_2, \gamma)$, ε is random error, and $C(X) = \mathbf{1}\{\omega^T X > \gamma\}$. In this setting, the test for the presence of a change-line is equivalent to testing the null hypothesis $H_0 : \alpha_0 = 0$, $\beta_0 = 0$. Under the null hypothesis, H_0, a cut-point γ does not exist anymore, and γ is identified only under



(a) Change in mean, under unequal variance assumption



(b) Change in standard deviation, under equal variance assumption

Figure 3.7: RLOWESS curves for the Gaussian kernel mean and standard deviation of the Toxicity data with two PCs. $\hat{\omega}$ and $\hat{\gamma}$ were obtained by change-line classification based on the subsample of size 1000. Dots denote observed toxicity activity and residual $\sqrt{(toxicity - \hat{\mu}(toxicity)_{Gaussian})^2}$ for mean and standard deviation, respectively. A solid curve denotes an RLOWESS regression line for the Gaussian kernel estimates with bandwidth calculated by Silverman's optimal bandwidth calculation for normal kernel.

the alternative hypothesis. This is not a standard testing problem where all parameters are identified under the null and alternative, and thus the traditional asymptotic optimality results of the Wald test and the likelihood ratio (LR) test no longer hold ([Andrews and Ploberger, 1994]; [DAVTES, 1977]; [Andrews, 2001]; [Kosorok and Song, 2007]). Researchers have proposed several hypothesis testing tools for testing for the presence of a change-point. The general idea is to compute test statistics, for example, $LR(\gamma)$ for each fixed change-point γ, then reject the null hypothesis if $\sup_{\gamma \in H} LR(\gamma; \zeta_0, \hat{\zeta}) > c$, where H is a parameter space of γ, and $\zeta_0, \hat{\zeta}$ are model parameters under the null and alternative hypotheses, respectively, and appear in both null and alternative hypotheses. A critical value c can be calculated based on the asymptotic distribution under the null hypothesis. Andrews and Ploberger [1994] and Andrews [2001] discussed assumptions for these types of testing for a change-point problem, but we do not discuss them in this dissertation.

We apply the same idea to conduct a hypothesis test for the presence of a change-line. In this study, we applied two test statistics, the sup functional and mean functional score statistics following Kosorok and Song [2007] which are computationally less intensive without estimating for (ω_0, γ_0). In this dissertation, a bootstrap resampling technique is utilized to obtain the empirical distribution of the test statistics without studying the asymptotic properties of the proposed statistics.

3.4.1 Set-up

Score functions and bootstrap estimators

For fixed $\varphi = (\omega, \gamma)$, the score function for $\zeta = (\alpha, \beta, \theta_1, \theta_2)$ in formula (3.4) is given as

$$U_\varphi \equiv (U_{\varphi,1}(\zeta), U_{\varphi,2}(\zeta), U_{\varphi,3}(\zeta), U_{\varphi,4}(\zeta))^T,$$

where

$$
\begin{aligned}
U_{\varphi,1}(\zeta) &= \sum_{i=1}^{n}(1-C(x_i))\left(\frac{y_i-(\theta_1+\alpha)}{\theta_2+\beta}\right)\\
U_{\varphi,2}(\zeta) &= -\frac{1}{2}\sum_{i=1}^{n}(1-C(x_i))\left(\frac{(\theta_2+\beta)-(y_i-(\theta_1+\alpha))^2}{(\theta_2+\beta)^2}\right)\\
U_{\varphi,3}(\zeta) &= \sum_{i=1}^{n}\left(C(x_i)\left(\frac{y_i-\theta_1}{\theta_2}\right)+(1-C(x_i))\left(\frac{y_i-(\theta_1+\alpha)}{\theta_2+\beta}\right)\right)\\
U_{\varphi,4}(\zeta) &= -\frac{1}{2}\sum_{i=1}^{n}\left(C(x_i)\left(\frac{\theta_2-(y_i-\theta_1)^2}{\theta_2^2}\right)+(1-C(x_i))\left(\frac{(\theta_2+\beta)-(y_i-(\theta_1+\alpha)^2)}{(\theta_2+\beta)^2}\right)\right).
\end{aligned}
$$

Following Kosorok and Song [2007], we define

$$
\hat{\zeta}_0^o \equiv \arg\max_{\theta_1,\theta_2}\mathbb{P}_n^o\ell(\theta;x,y)=(0,0,\hat{\theta}_1^o,\hat{\theta}_2^o),
$$

where $\mathbb{P}_n^o f(x)\equiv n^{-1}\sum_{i=1}^n\epsilon_i f(x_i)$ for some positive random variables ϵ and a function f. Under H_0, note that $\alpha=0=\beta$, and we do not have to compute score statistics for the change-line parameters $\varphi=(\omega,\gamma)$. Bootstrap estimators for θ_1 and θ_2 under H_0 are given as

$$
(\hat{\theta}_1^o,\hat{\theta}_2^o) = \arg\max_{\theta_1,\theta_2} n^{-1}\sum_{i=1}^{n}k_i^o\left(-\frac{1}{2}\log\theta_2-\frac{(y_i-\theta_1)^2}{2\theta_2}\right).
$$

Here $k_i^o=k_i/\bar{k}_n$, $i=1,\ldots,n$ are standardized weights, where $\bar{k}_n=n^{-1}\sum_{i=1}^n k_i$ and k_is are n i.i.d positive random variables with mean $0<\mu_k<\infty$, variance $0<\sigma_k^2<\infty$, and satisfying $\int_0^\infty\sqrt{P(k_1>u)}du<\infty$. In this study, random variables were generated from the exponential distribution with $\mu_k=1$. By the property of k_i^o, bootstrap estimators with weights $\hat{\zeta}_0^o=(\hat{\theta}_1^o,\hat{\theta}_2^o)$ are given as

$$
\begin{aligned}
\hat{\theta}_1^o &= n^{-1}\sum_{i=1}^{n}k_i^o y_i\\
\hat{\theta}_2^o &= n^{-1}\sum_{i=1}^{n}k_i^o(y_i-\hat{\theta}_1^o)^2. \qquad (3.5)
\end{aligned}
$$

Using the same weights $k_1^o, \ldots, k_n^o$, score statistics at each φ are computed by

$$\begin{aligned}
\hat{S}_1^o(\varphi) &\equiv \sqrt{n}\mathbb{P}_n^o(U_{\varphi,1}(\hat{\zeta}_0^o), U_{\varphi,2}(\hat{\zeta}_0^o)) \\
&= \sqrt{n}n^{-1}\sum_{i=1}^{n} k_i^o (U_{\varphi,1}(\hat{\zeta}_0^o), U_{\varphi,2}(\hat{\zeta}_0^o))^T \\
&= \begin{pmatrix} n^{-1/2}\sum_{i=1}^{n} k_i^o (1 - C(x_i)) \dfrac{(y_i - \hat{\theta}_1^o)}{\hat{\theta}_2^o} \\ -\frac{1}{2} n^{-1/2} \sum_{i=1}^{n} k_i^o (1 - C(x_i)) \left(\dfrac{\hat{\theta}_2^o - (y_i - \hat{\theta}_1^o)^2}{\hat{\theta}_2^{o,2}} \right) \end{pmatrix}_{2\times 1}. \qquad (3.6)
\end{aligned}$$

Bootstrap estimating procedure

1. First, weight samples $k^{o(b)}, b = 1, \ldots, B$ are generated from the exponential distribution with mean of 1, where B is a large number of bootstrap replications, and each $k^{o(b)}$ consists of n elements.

2. For b^{th} weight sample $k^{o(b)} = (k_1^{o(b)}, \ldots, k_n^{o(b)})$ and for fixed $\varphi = \{(\omega_1^{(i)}, \omega_2^{(i)}, \gamma^{(i,j)})_{i=1,\ldots,K, j=1,\ldots,n-1}\}$, we calculate $\hat{\theta}_1^{o(b)}$ and $\hat{\theta}_2^{o(b)}$ as described in (3.5).

3. Using the same weight samples, we calculate $\{\hat{S}_1^o(\varphi^{(i,j)}), \varphi^{(i,j)} = (\omega_1^{(i)}, \omega_2^{(i)}, \gamma^{(i,j)})\}$ as given in (3.6). After conducting Step 1 to Step 3, we have $\{\hat{S}_{1,1}^o, \ldots, \hat{S}_{1,B}^o\}$, and each $\hat{S}_{1,b}^o$ is a $2 \times (K \times (n-1))$ matrix, where $K \le n(n-1)/2$.

4. For each $\varphi = (\omega, \gamma)$, compute bootstrap mean and bootstrap variance by

$$\begin{aligned}
\hat{\mu}_n(\varphi) &= B^{-1}\sum_{b=1}^{B} \hat{S}_{1,b}^o(\varphi)|_{2\times 1} \\
\hat{V}_n(\varphi) &= B^{-1}\sum_{l=1}^{B} \{\hat{S}_{1,b}^o(\varphi) - \hat{\mu}_n(\varphi)\}\{\hat{S}_{1,b}^o(\varphi) - \hat{\mu}_n(\varphi)\}^T|_{2\times 2}. \qquad (3.7)
\end{aligned}$$

5. Compute two test statistics by

$$\begin{aligned}\hat{T}_n &\equiv \sup_{\varphi}\{\hat{S}_1(\varphi)^T\hat{V}_n^{-1}(\varphi)\hat{S}_1(\varphi)\} \\ \tilde{T}_n &\equiv \int_{\varphi}\{\hat{S}_1(\varphi)^T\hat{V}_n^{-1}(\varphi)\hat{S}_1(\varphi)\}d\varphi = \sum_{\varphi}\{\hat{S}_1(\varphi)^T\hat{V}_n^{-1}(\varphi)\hat{S}_1(\varphi)\}.\end{aligned}$$

$\hat{T}_n$ is a sup functional score statistic, where the supremum is taken over $\varphi = (\omega, \gamma)$ while $\tilde{T}_n$ is a mean score test statistic calculated by integration over all $\varphi = (\omega, \gamma)$. Here $\hat{S}_1(\varphi) \equiv \sqrt{n}\mathbb{P}_n(U_{\varphi,1}(\hat{\zeta}_0), U_{\varphi,2}(\hat{\zeta}_0))^T$, where $\hat{\zeta}_0 \equiv \arg\max_{\theta_1,\theta_2} L_n$ under H_0. $\hat{\zeta}_0 = (0, 0, \hat{\theta}_1, \hat{\theta}_2)$, where $\hat{\theta}_1 = \bar{y}$, and $\hat{\theta}_2 = n^{-1}\sum_{i=1}^n (y_i - \bar{y})^2$. Note that $\hat{V}_n(\varphi)$ is the same variance estimate given in (3.7), and $\hat{S}_1(\varphi)$ is computed by

$$\hat{S}_1(\varphi) = \begin{pmatrix} n^{-1/2}\sum_{i=1}^n (1 - C(x_i))\left(\frac{y_i - \hat{\theta}_1}{\hat{\theta}_2}\right) \\ -\frac{1}{2}n^{-1/2}\sum_{i=1}^n (1 - C(x_i))\left(\frac{\hat{\theta}_2 - (y_i - \hat{\theta}_1)^2}{\hat{\theta}_2^2}\right)\end{pmatrix}. \tag{3.8}$$

6. To estimate a critical value, we compute the standardized bootstrap test statistic for each b, $b = 1, \ldots, B$ by

$$\begin{aligned}\hat{T}^o_{n,b} &\equiv \sup_{\varphi}\{(\hat{S}^o_{1,b}(\varphi) - \hat{\mu}_n(\varphi))^T\hat{V}_n^{-1}(\varphi)(\hat{S}^o_{1,b}(\varphi) - \hat{\mu}_n(\varphi))\} \\ \tilde{T}^o_{n,b} &\equiv \int_{\varphi}\{(\hat{S}^o_{1,b}(\varphi) - \hat{\mu}_n(\varphi))^T\hat{V}_n^{-1}(\varphi)(\hat{S}^o_{1,b}(\varphi) - \hat{\mu}_n(\varphi))\}d\varphi.\end{aligned}$$

7. For a test of size ϕ, we compute the test statistics $\hat{T}_n$ and $\tilde{T}_n$ with the $(1-\phi)^{th}$ quantile of the corresponding B standardized bootstrap statistics. We reject $H_0 : \alpha_0 = 0 = \beta_0$ if $\hat{T}_n > \hat{T}^o_{n,(1-\phi)}$ or if $\tilde{T}_n > \tilde{T}^o_{n,(1-\phi)}$. In this study, the targeted size of ϕ was 0.05.

3.4.2 Preliminary simulation study

While conducting simulation studies for the hypothesis testing as described in section 3.4.1, we experienced a computational challenge. To obtain the empirical distribution of the test statistics, we need much larger memory than that for the estimating procedure. Due to computational difficulty, we ran simulation studies in limited situations with small numbers of observations and/or small numbers of bootstrap replications. We tried three different methods of conducting hypothesis tests. First, we calculated test statistics for all $\varphi = (\omega, \gamma)$ in the whole range of ω and γ (denoted by full search). In this case, we can run simulations on the small data set with a relatively small number of bootstrap replications. Second, score statistics were computed at all values for ω but at randomly selected γ (random search). This allowed us to increase the number of bootstrap replications, but we had additional randomness in the bootstrap random weights. To remove randomness, we used a grid searching for both ω and γ (grid search). After sorting ω and γ, we calculated score statistics at the points on a fixed grid constructed by (ω_i, γ_j), where $i = 1, 3, \ldots$ (50% of the whole ω set) and $j = 1, 5, 9, \ldots$ (25% of the whole γ set).

Two different scenarios were simulated: First, two latent subgroups had different means and variances, so one group followed $\mathbf{N}(0, 1)$, and the other group followed $\mathbf{N}(2, 4)$. Second, we simulated data for the null hypothesis, that is, two subgroups had the same mean and variance, $\mathbf{N}(2, 4)$.

The simulation results are given in Table 3.8. Using the full search method, the number of bootstrap replications (B) was 100 with sample size (n) of 100. Under the alternative (two heterogeneous subgroups exist), 98% and 100% of the null hypotheses were rejected over 100 MC simulations by sup score test statistics and mean

score statistics, respectively. The number of rejected null hypotheses were higher than the target of 95%. Under the null (two subgroups have the same distribution), only 2% of the null hypotheses were rejected by sup score test statistics, but 23% of the null hypotheses were rejected by mean score test statistics. Using the random search method, the number of bootstrap replications was 250 with sample size 100, and 98.44% and 98% of the null hypotheses were rejected by sup score test statistics and mean score test statistics, respectively. Under the null, only 1.8% of null hypotheses were rejected by the sup score test statistics, but 14.6% of the null hypotheses were rejected by mean score test statistics. Using the grid searching method, the number of bootstrap replications was 500 with sample size 150, and 100% of the null hypotheses were rejected by sup and mean score test statistics under the alternative. Under the null, only 3% of the null hypotheses were rejected by sup score test statistics while 73% of the null hypotheses were rejected by mean score test statistics.

In our experiment, both sup and mean score test statistics tend to reject the null hypothesis more frequently than the target level under the alternative. Under the null, the mean score test statistic tends to reject too many while the sup score test statistic rejects too few. Again, this experiment was conducted in a very limited setting with a small number of observations due to computational burden. Our results suggest that we need further investigation with a more computationally efficient method to calculate test statistics for fixed change-line parameters.

3.4.3 Example: Chemical toxicity data

We conducted a hypothesis test for the presence of a change-line in the chemical toxicity data using the full search method. We ran 100 MC simulations, and for

Table 3.8: The number of rejections of the null hypothesis are presented in percentage scale ($\frac{\#\text{of rejection}}{\text{total \# of MC simulations}} \times 100$). Sup score and mean score denote sup score test statistics and mean score test statistics, respectively.

	Under the alternative			Under the null		
Method	Setting	Sup score	Mean score	Setting	Sup score	Mean score
Full search	n=100 B=100 MC=100	98%	100%	n=100 B=100 MC=100	2%	23%
Random search	n=100 B=250 MC =500 50% of γ	98.44%	98%	n=100 B=250 MC =500 50% of γ	1.8%	14.6%
Grid search	n=150 B=500 MC=100 25% of ω 50% of γ	100%	100%	n=150 B=500 MC=100 25% of ω 50% of γ	3%	73%

n: sample size
B: the number of bootstrap replications for each MC simulations
MC: the number of MC simulations

each simulation, 100 chemical compounds were randomly selected without replacement. Sup score test statistics and mean score test statistics were calculated using 100 bootstrap replications. Sup and mean score test statistics rejected the null hypothesis 98 times and 100 times, respectively. Note that this result is consistent with a graphical examination although this test was conducted on a very small subset of data with a small number of bootstrap replications.

Chapter 4

Asymptotic properties in the change-line regression model

In this chapter, we study asymptotic properties including the consistency and the rates of convergence of the estimators obtained by the two-stage estimating procedure proposed in Chapter 3.1, focusing on the change-line regression model.

4.1 Model and assumptions

We observe n independent and identically distributed (i.i.d.) realizations of the random triple (Y, Z, X) in a probability space $(\mathcal{X}, \mathcal{A}, P)$ such that

$$Y = \mathbf{1}\{\omega^T X - \gamma > 0\}\beta^T Z + \mathbf{1}\{\omega^T X - \gamma \leq 0\}\delta^T Z + \epsilon, \tag{4.1}$$

where $\omega = (\omega_1, \omega_2) \in \mathcal{S}^2 = \{(\omega_1, \omega_2) \in \mathbb{R}^2 : \|\omega\| = 1\}$, and $\gamma \in [a, b] \in \mathbb{R}$. We assume that all assumptions and conditions that we made in Chapter 3.1 hold for (4.1).

Summary of conditions

C1. The variable $X \in \mathbb{R}^2$ has a strictly bounded and positive density over $[a, b]$ with

$P(\omega_0^T X < a) > 0$ and $P(\omega_0^T X > b) > 0$.

C2. $Z \in \mathbb{R}^q$ with $P\|Z\|^2 < \infty$.

C3. X and Z are independent of the random error ϵ, and $P\epsilon = 0$, $\sigma^2 = P\epsilon^2 < \infty$.

C4. Change-line parameters $\varphi = (\omega, \gamma) \in \mathcal{S}^2 \times [a, b]$, so $\hat{\omega}$ and $\hat{\gamma}$ exist. Also, model parameter ζ ranges over some known compact set H_1 in $\mathbb{R}^P$ so that ζ is bounded, and hence $\hat{\zeta}$ exists.

C5. $\beta_0 \neq \delta_0$, and a change-line exists at (ω_0, γ_0) for identifiability.

A1. There exists an open set $A \in \mathbb{R}^2$ such that the density of X on the closure $\bar{A}$ is bounded below and $\omega_0^T X = \gamma_0$ for some $X \in A$ (assumption **A**).

A2. The density of $\omega_0^T X - \gamma$ is assumed to be positive in a neighbor of A (assumption **B**).

Under the given regularity assumptions, the goal is to estimate θ through the least squares method, and this is the same as finding an M-estimator that maximizes $M_n(\theta) = \mathbb{P}_n m_\theta$, where $\mathbb{P}_n f(x) = n^{-1} \sum_{i=1}^n f(x_i)$ and

$$m_\theta = -(y - \beta^T Z \mathbf{1}\{\omega^T X - \gamma > 0\} - \delta^T Z \mathbf{1}\{\omega^T X - \gamma \leq 0\})^2. \tag{4.2}$$

Let $\hat{\theta}_n$ be a maximizer of $M_n(\theta)$, where $\hat{\theta}_n \equiv (\hat{\zeta}, \hat{\varphi})$, $\hat{\varphi} = (\hat{\omega}, \hat{\gamma})$, $\hat{\zeta} = (\hat{\beta}_1, \ldots, \hat{\beta}_p, \hat{\delta}_1, \ldots, \hat{\delta}_p)$. In this dissertation, we study the consistency of the M-estimator $\hat{\theta}_n$ for the change-line regression model (4.1), and the rates of convergence for the estimators of model parameters and change-line parameters.

4.2 Consistency

To study consistency, we use the Argmax theorem introduced in Chapter 7.1. The Argmax theorem is a special case of a continuous mapping theorem [Van Der Vaart and Wellner, 1996]. In the present setting, θ_0 is a fixed number, so hence it is tight. Also, by definition of $\hat{\theta}_n$ that maximizes M_n over θ, the condition of nearly maximization is satisfied. The uniform convergence of M_n to M in $\ell^\infty(K)$ and the uniform tightness of $\hat{\theta}_n$ are related to the compactness, where $M(\theta) = Pm_\theta \equiv \int_\chi m_\theta(x)dP(x)$, and $\ell^\infty(K)$ is the space of bounded functionals on K. The upper semicontinuity and uniqueness of maximum of M at θ_0 are related to the identifiability.

lemma 1. *(consistency of $\hat{\theta}_n$) Under the regularity conditions given in Chapter 3.1 and 4.1, $\hat{\theta}_n \to \theta_0$ in probability.*

4.2.1 Compactness conditions

1. $M_n \rightsquigarrow M$ **in** $\ell^\infty(K)$ **for every compact** $K \subset H$**.**

To verify that $M_n \rightsquigarrow M$ in $\ell^\infty(K)$ for every compact $K \subset H$, we show that the class of functions $\mathcal{F}_K \equiv \{m_\theta : \theta \in K\}$ is Glivenko-Cantelli. We can rewrite m_θ as

$$m_\theta = -\begin{pmatrix} \beta_0^T Z \mathbf{1}\{\omega_0^T X - \gamma_0 > 0\} + \delta_0^T Z \mathbf{1}\{\omega_0^T X - \gamma_0 \le 0\} + \epsilon \\ -\beta^T Z \mathbf{1}\{\omega^T X - \gamma > 0\} - \delta^T Z \mathbf{1}\{\omega^T X - \gamma \le 0\} \end{pmatrix}^2.$$

In a similar way to showing consistency in the one-dimensional change-point problem of Kosorok [2008b], we decompose the range of γ and γ_0 into four intervals separately, $(\{\omega^T X \le \gamma_0 \wedge \gamma\}, \{\gamma < \omega^T X \le \gamma_0\}, \{\gamma_0 < \omega^T X \le \gamma\}, \{\omega^T X > \gamma_0 \vee \gamma\}) \times (\{\omega_0^T X \le$

$\gamma_0 \wedge \gamma\}, \{\gamma < \omega_0^T X \le \gamma_0\}, \{\gamma_0 < \omega_0^T X \le \gamma\}, \{\omega_0^T X > \gamma_0 \vee \gamma\})$. Now, m_θ can be written as

$$\begin{aligned} m_\theta = &-(\epsilon - (\delta - \delta_0)^T Z)^2 \times A_1(\omega, \omega_0, \gamma, \gamma_0; X) - (\epsilon - (\delta - \beta_0)^T Z)^2 \times A_2(\omega, \omega_0, \gamma, \gamma_0; X) \\ &- (\epsilon - (\beta - \delta_0)^T Z)^2 \times A_3(\omega, \omega_0, \gamma, \gamma_0; X) - (\epsilon - (\beta - \beta_0)^T Z)^2 \times A_4(\omega, \omega_0, \gamma, \gamma_0; X), \end{aligned} \tag{4.3}$$

where

$$\begin{aligned} A_1(\omega, \omega_0, \gamma, \gamma_0; X) = &\mathbf{1}\{\omega^T X \le \gamma_0 \wedge \gamma\}\mathbf{1}\{\omega_0^T X \le \gamma_0 \wedge \gamma\} + \mathbf{1}\{\omega^T X \le \gamma_0 \wedge \gamma\}\mathbf{1}\{\gamma < \omega_0^T X \le \gamma_0\} \\ &+ \mathbf{1}\{\gamma_0 < \omega^T X \le \gamma\}\mathbf{1}\{\gamma < \omega_0^T X \le \gamma_0\} + \mathbf{1}\{\gamma_0 < \omega^T X \le \gamma\}\mathbf{1}\{\omega_0^T X \le \gamma_0 \wedge \gamma\}, \\ A_2(\omega, \omega_0, \gamma, \gamma_0; X) = &\mathbf{1}\{\omega^T X \le \gamma_0 \wedge \gamma\}\mathbf{1}\{\gamma_0 < \omega_0^T X \le \gamma\} + \mathbf{1}\{\omega^T X \le \gamma_0 \wedge \gamma\}\mathbf{1}\{\omega_0^T X > \gamma_0 \vee \gamma\} \\ &+ \mathbf{1}\{\gamma_0 < \omega^T X \le \gamma\}\mathbf{1}\{\gamma_0 < \omega_0^T X \le \gamma\} + \mathbf{1}\{\gamma_0 < \omega^T X \le \gamma\}\mathbf{1}\{\omega_0^T X > \gamma_0 \vee \gamma\}, \\ A_3(\omega, \omega_0, \gamma, \gamma_0; X) = &\mathbf{1}\{\gamma < \omega^T X \le \gamma_0\}\mathbf{1}\{\omega_0^T X < \gamma_0 \wedge \gamma\} + \mathbf{1}\{\gamma < \omega^T X \le \gamma_0\}\mathbf{1}\{\gamma < \omega_0^T X \le \gamma_0\} \\ &+ \mathbf{1}\{\omega^T X > \gamma_0 \vee \gamma\}\mathbf{1}\{\omega_0^T X < \gamma_0 \wedge \gamma\} + \mathbf{1}\{\omega^T X > \gamma_0 \vee \gamma\}\mathbf{1}\{\gamma < \omega_0^T X \le \gamma_0\}, \\ A_4(\omega, \omega_0, \gamma, \gamma_0; X) = &\mathbf{1}\{\gamma < \omega^T X \le \gamma_0\}\mathbf{1}\{\omega_0^T X > \gamma_0 \vee \gamma\} + \mathbf{1}\{\omega^T X > \gamma_0 \vee \gamma\}\mathbf{1}\{\gamma_0 < \omega_0^T X \le \gamma\} \\ &+ \mathbf{1}\{\omega^T X > \gamma_0 \vee \gamma\}\mathbf{1}\{\omega_0^T X > \gamma_0 \vee \gamma\} + \mathbf{1}\{\gamma < \omega^T X \le \gamma_0\}\mathbf{1}\{\gamma_0 < \omega_0^T X \le \gamma\}. \end{aligned} \tag{4.4}$$

Claim 1. $\{(\epsilon - (\delta - \delta_0)^T Z)^2 : \theta \in K, P\epsilon = 0, P\epsilon^2 < \infty\}$ *is Glivenko-Cantelli.*

Proof. *Claim 1*

Note that $(\epsilon - (\delta - \delta_0)^T Z)^2 = \epsilon^2 - 2(\delta - \delta_0)^T Z\epsilon + ((\delta_0 - \delta)^T Z)^2$, *and we show that each component of the formula is Glivenko-Cantelli. First, for every function* f *in* $\mathcal{F}_1 \equiv \{f = (\delta - \delta_0)^T Z : Z \in \mathbb{R}^q, \delta, \delta_0 \in K\}$,

$$\begin{aligned} \sup_{f \in \mathcal{F}_1} (P_n - P) f &= \sup_\delta |(\delta - \delta_0)^T (P_n - P) Z| \\ &\le \sup_\delta \|\delta - \delta_0\| \|(P_n - P) Z\| \\ &\le K_1 \times \|(P_n - P) Z\| \longrightarrow 0, \end{aligned}$$

where K_1 denotes a constant through the compactness of K, and Z is an i.i.d. random variable in $\mathbb{R}^q$, so $(P_n - P)Z \xrightarrow{a.s.} 0$ by the Strong law of large numbers. Therefore, we conclude that $\sup_{f \in \mathcal{F}_1} |(P_n - P)f| \xrightarrow{a.s.} 0$ for every compact $K \subset H$, and thus the class $\mathcal{F}_1$ is (strong) Glivenko-Cantelli.

Similarly, for every $f \in \mathcal{F}_2 = \{f = (\delta - \delta_0)^T Z\epsilon : Z \in \mathbb{R}^q, \delta, \delta_0 \in K, P\epsilon = 0, P\epsilon^2 < \infty\}$,

$$\begin{aligned}\sup_{f \in \mathcal{F}_2} (P_n - P)f &\leq \sup_{\delta} \|\delta - \delta_0\| \|(P_n - P)Z\epsilon\| \\ &\leq K_2 \times \|(P_n - P)Z\epsilon\| \longrightarrow 0.\end{aligned}$$

Since $Z \perp\!\!\!\perp \epsilon$, $P_n Z\epsilon \to PZP\epsilon = 0$. Thus $\sup_{f \in \mathcal{F}_2} |(P_n - P)f| \xrightarrow{a.s.} 0$ for every compact $K \subset H$, and hence this class $\mathcal{F}_2$ is Glivenko-Cantelli. Assuming that $P\epsilon = 0$ and $P\epsilon^2 < \infty$, $\{\epsilon^2\}$ is P-Glivenko-Cantelli by the Strong law of large numbers. Combining with the property of Glivenko-Cantelli preservation stating that $\mathcal{F} + \mathcal{G}$ and $\mathcal{F}\mathcal{G}$ are Glivenko-Cantelli whenever $\mathcal{F}$ and $\mathcal{G}$ are Glivenko-Cantelli, we conclude that $\{(\epsilon - (\delta - \delta_0)^T Z)^2 : \delta, \delta_0 \in K, P\epsilon = 0, P\epsilon^2 < \infty, Z \in \mathbb{R}^2\}$ is Glivenko-Cantelli. Thus Claim 1 is verified.$\square$

Claim 2. *$\{\mathbf{1}\{\omega^T X \leq \gamma_0 \wedge \gamma\} : \theta \in K, \gamma, \gamma_0 \in [a, b]\}$ is Glivenko-Cantelli.*

To prove Claim 2, we show that $\{\mathbf{1}\{\omega^T X \leq \gamma_0 \wedge \gamma\} : \theta \in K, \gamma, \gamma_0 \in [a, b]\}$ is a Donsker class (so hence Glivenko-Cantelli) by Theorem 3 in Chapter 7.1. For simplicity, let $\gamma_0 \wedge \gamma = \gamma \in [a, b]$. Then it suffices to show that the class of functions $\mathcal{F} \equiv \{\mathbf{1}\{\omega^T X - \gamma \leq 0\} : \omega \in \mathcal{S}^2, X \in \mathbb{R}^2, \gamma \in [a, b]\}$ satisfies the following conditions:

1. $J(1, \mathcal{F}, L_2) < \infty$.
2. $\mathcal{F}, \mathcal{F}_\delta$, and $\mathcal{F}_\infty^2$ are pointwise measurable (PM).

3. $P^*F^2 < \infty$ for envelope function F.

$J(1, \mathcal{F}, L_2)$ is the uniform entropy integral as given in Chapter 7.1. Definitions of $\mathcal{F}, \mathcal{F}_\delta$, and $\mathcal{F}_\infty^2$ are provided in 3, Chapter 7.1. Outer probability P^* is defined as the infimum over all $P(B)$ with $A \subset B \subset \Omega$ and B is a Borel measurable set. Here is the basic idea to verify that $\mathcal{F}$ satisfies three conditions above. First, Claim 2-1 below shows that $\mathcal{F}$ is a Vapnik-Červonenkis class (VC-class) with VC-index $V(\mathcal{F}) \leq 5$, and hence $\mathcal{F}$ has a bounded uniform entropy integral (BUEI) with an envelope function F by Theorem 4 in Chapter 7.1. Second, $\mathcal{F}$, $\mathcal{F}_\delta$, and $\mathcal{F}_\infty^2$ are PM by Lemma 2 in Chapter 7.1. The last condition is also satisfied because $\mathcal{F}$ has a bounded envelope function by the property of the indicator function. Combining all three results, we can conclude $\mathcal{F}$ is P-Donsker, and thus this class is P-Glivenko-Cantelli.

Claim 2.1. $\{\mathbf{1}\{\omega^T X - \gamma \leq 0\} : \theta \in K, \gamma \in [a,b]\}$ is a VC-class with VC-index $V(\mathcal{F}) \leq 5$.

Proof. *Claim 2.1. We verify Claim 2.1 in three steps:*

2.1-A. $\{\omega^T X - \gamma, \omega \in \mathcal{S}^2, X \in \mathbb{R}^2, \gamma \in [a,b]\}$ *is VC-subgraph with VC-index* ≤ 5.

2.1-B. *The set* $\{\omega^T X - \gamma > 0\}$ *is VC-class with VC-index* ≤ 5.

2.1-C. $\mathbf{1}\{\omega^T X - \gamma \leq 0\}$ *is VC-class with VC-index* ≤ 5.

For 2.1-A, we define a function $f : \mathcal{X} \to \mathbb{R}$ *by* $f_\omega(X) = \langle \omega, X \rangle$, *where* $\omega = (\omega_1, \omega_2, \omega_3) \in \mathbb{R}^3$, *and* $X = (X_1, X_2, X_3) \in \mathbb{R}^3$. *Then* $\{\omega^T X\}$ *is in a three-dimensional vector space, so* $\{\omega^T X\}$ *is a VC-subgraph with VC-index less than or equal to 5 by Lemma 4 in chapter 7.1. Note that* $\{\omega^T X - \gamma, \omega \in \mathcal{S}^2, \gamma \in [a,b], X \in \mathbb{R}^2\}$ *is a subset of* $\{\omega^T X\}$ *with* $(\omega_1, \omega_2) \in \mathcal{S}^2$, $\omega_3 = \gamma$, *and* $X_3 = -1$. *Thus this is also a VC-subgraph with VC-index less than or equal to 5. For 2.1-B, by Lemma 7-3 in chapter 7.1, the set* $\{\omega^T X - \gamma > 0\}$ *is a VC-class of sets because* $\{\omega^T X - \gamma\}$ *is a VC-subgraph. Since the complement of*

a VC-class of sets is also a VC-class of sets, $\{\omega^T X - \gamma \le 0\}$ *is a VC-class of sets with VC-index less than or equal to 5. 2.1-C is easily verified by Lemma 6 in chapter 7.1 which states that the class of indicator functions of a VC-class of sets is again a VC-subgraph. Therefore, we conclude that* $\mathbf{1}\{\omega^T X - \gamma \le 0, \omega \in \mathcal{S}^2, \gamma \in [a,b], X \in \mathbb{R}^2\}$ *is a VC-subgraph with VC-index less than or equal to 5. Now, by Theorem 4 in chapter 7.1, there exists a universal constant* $K_3 < \infty$ *such that for any VC-class of measurable function* $\mathcal{F}$ *with an integrable envelope function* $F = 1$ *and* $r = 2$*, for any probability measure* Q *with* $\|F\|_{Q,2} > 0$*, and for any* $0 < \varepsilon < 1$*,*

$$\begin{aligned} N(\varepsilon \|F\|_{Q,2}, \mathcal{F}, L_2(Q)) &\le K_3 V(\mathcal{F})(4e)^{V(\mathcal{F})} \Big(\frac{2}{\varepsilon}\Big)^{2(V(\mathcal{F})-1)} \\ &= 5K_3(4e)^5 \Big(\frac{2}{\varepsilon}\Big)^{2(5-1)} \\ &\lesssim K_3 \Big(\frac{1}{\varepsilon}\Big)^8. \end{aligned}$$

Therefore, $\log N(\varepsilon \|F\|_{Q,2}, \mathcal{F}, L_2(Q)) \lesssim \log \frac{1}{\varepsilon}$*, and finally we verify that* $\mathcal{F}$ *satisfies the first condition for Claim 2. That is,* $J(1, \mathcal{F}, L_2) < \infty$ *(BUEI).*

Claim 2.2. $\mathcal{F}$, $\mathcal{F}_\delta$, and $\mathcal{F}_\infty^2$ are P-measurable (PM) for every $\delta > 0$.

Proof. *Claim 2.2 can be verified by following the proof of Lemma 8.12 of Kosorok [2008b]. First, we consider this problem in* $\mathcal{X}_M = \{(X_1, X_2) \in \mathbb{R}^2, \|X\| < M\}$ *for some fixed* $M < \infty$*. Then, we can show that there exists a countable subset* $\mathcal{G} \subset \mathcal{F}$ *such that for every* $f \in \mathcal{F}$*, there exists a sequence* $\{g_m\} \in \mathcal{G}$ *with* $g_m(x) \to f(x)$ *for every* $x \in \mathcal{X}_M$*. For example,* $\mathcal{G} = \{\mathbf{1}\{\omega^T X \le \gamma\} : \omega \in \{(\omega_1, \omega_2) \in \mathbb{Q}^2, \|\omega\| = 1\}, X \in \mathcal{X}_M, \gamma \in \mathbb{Q}\}$*, where* $\mathbb{Q}$*'s are the rationals, and* $g_m = \{\omega^T X \le \gamma + r_m\}$*. Note that* $r_m = \gamma_m - \gamma + (\omega - \omega_m)^T X$*,* $\omega_m \in \mathbb{Q}^2$ *satisfying* $\|\omega_m - \omega\| \le \dfrac{1}{2mM}$*,* $\gamma_m \in (\gamma + \dfrac{1}{2m}, \gamma + \dfrac{1}{m}) \subset \mathbb{Q}$ *for any* $X \in \mathcal{X}_M$*. Then* $\mathcal{F}$ *is PM by Definition 7 in Chapter 7.1. Once we show* $\mathcal{F}$ *is PM,* $\mathcal{F}_\delta$ *and* $\mathcal{F}_\infty^2$ *can be verified as PM on* $\mathcal{X}_M$ *in a similar way to verifying that* $\mathcal{F}$ *is PM. Now, let* $J_M(x_1, \ldots, x_n) = \mathbf{1}\{\max_{1 \le i < n} \|x_i\| \le M\}$*. Since* M *is arbitrary, the map*

$$(x_1, x_2, \ldots, x_n) \mapsto \| \sum_{i=1}^{n} e_i f(x_i) \|_{\mathcal{H}} J_M(x_1, \ldots, x_n) \tag{4.5}$$

is measurable for every n-tuple $(e_1, \ldots, e_n) \in \mathbb{R}^n$*, every* $M < \infty$*, and with* $\mathcal{H}$ *being replaced with* $\mathcal{F}, \mathcal{F}_\delta$*, or* $\mathcal{F}_\infty^2$*. Now, for any* $(x_1, \ldots, x_n) \in \mathcal{X}^n, J_M(x_1, \ldots, x_n) = 1$ *for all* M *large enough. Thus, (4.5) is also measurable after replacing* J_M *with its pointwise limit 1, and* $\mathcal{F}$*,* $\mathcal{F}_\delta$*, and* $\mathcal{F}_\infty^2$ *are PM for any measure* P *on* $\mathcal{X}$*. This verifies Claim 2.2. Detailed proofs can be found in Chapter 7.2.*

In Claim 2.1, we verified that $\mathcal{F} = \{\mathbf{1}\{\omega^T X - \gamma \le 0\} : \omega \in \mathcal{S}^2, \gamma \in [a, b], X \in \mathbb{R}^2\}$ is VC-class with VC-index less than or equal to 5, which implies $J(1, \mathcal{F}, L_2) < \infty$. From Claim 2.2, we also showed that $\mathcal{F}, \mathcal{F}_\delta$, and $\mathcal{F}_\infty^2$ are PM for any measure P on $\mathcal{X}$. Since $\mathbf{1}\{\omega^T X - \gamma \le 0\} \le F$ with $F = \mathbf{1}\{\mathcal{X}\}$, and also $\int_{\mathcal{F}} F^2 dp = P(\mathcal{X}) < \infty$, $\mathcal{F}$ has a square integrable envelope function. Therefore, by Theorem 3, Chapter 7.1, we conclude that $\mathcal{F} = \{\mathbf{1}\{\omega^T X \le \gamma \wedge \gamma_0\} : \omega \in \mathcal{S}^2, \gamma \in [a, b], X \in \mathbb{R}^2\}$ is P-Donsker, so hence this class of functions is P-Glivenko-Cantelli. This verifies Claim 2.□

In a way similar to the verification of Claim 2, we can show that $\mathbf{1}\{\omega_0^T X \le \gamma \wedge \gamma_0\}$, $\mathbf{1}\{\gamma < \omega_0^T X \le \gamma_0\}$, $\mathbf{1}\{\gamma_0 < \omega^T X \le \gamma\}$, and $\mathbf{1}\{\gamma_0 < \omega_0^T X \le \gamma\}$ are P-Donsker classes. By the property of Donsker preservation, all products and sums of Donsker classes with bounded envelope functions are still Donsker, thus $A_1(\omega, \omega_0, \gamma, \gamma_0; X)$ of (4.4) is a Donsker class. For the same reasons, $A_2(\omega, \omega_0, \gamma, \gamma_0; X)$, $A_3(\omega, \omega_0, \gamma, \gamma_0; X)$, and $A_4(\omega, \omega_0, \gamma, \gamma_0; X)$ of (4.4) are Donsker. Combining Claim 1 with Claim 2, $-(\epsilon - (\delta - \delta_0)^T Z)^2 \times A_1(\omega, \omega_0, \gamma, \gamma_0; X)$ is a Donsker class by the property of Donsker preservation. We have similar results for the other three terms of (4.3). Finally, we conclude that m_θ is a Donsker class, and hence that it is also Glivenko-Cantelli. That is, $M_n \rightsquigarrow M$ in $\ell^\infty(K)$ for every compact $K \subset H$.□

2. M has a unique maximum at θ_0.

In (4.4), $A_1 \Leftrightarrow \mathbf{1}\{\omega^T X \le \gamma\}\mathbf{1}\{\omega_0^T X \le \gamma_0\}$, $A_2 \Leftrightarrow \mathbf{1}\{\omega^T X \le \gamma\}\mathbf{1}\{\omega_0^T X > \gamma_0\}$, $A_3 \Leftrightarrow \mathbf{1}\{\omega^T X > \gamma\}\mathbf{1}\{\omega_0^T X \le \gamma_0\}$, and $A_4 \Leftrightarrow \mathbf{1}\{\omega^T X > \gamma\}\mathbf{1}\{\omega_0^T X > \gamma_0\}$. Therefore,

$$M(\theta) - M(\theta_0) = -P\begin{pmatrix} ((\delta-\delta_0)^T Z)^2\mathbf{1}\{\omega^T X \le \gamma\}\mathbf{1}\{\omega_0^T X \le \gamma_0\} \\ +((\delta-\beta_0)^T Z)^2\mathbf{1}\{\omega^T X \le \gamma\}\mathbf{1}\{\omega_0^T X > \gamma_0\} \\ +((\delta_0-\beta)^T Z)^2\{\omega^T X > \gamma\}\mathbf{1}\{\omega_0^T X \le \gamma_0\} \\ +((\beta_0-\beta)^T Z)^2\mathbf{1}\{\omega^T X > \gamma\}\mathbf{1}\{\omega_0^T X > \gamma_0\} \end{pmatrix}. \quad (4.6)$$

If either $\delta \neq \delta_0$ or $\beta \neq \beta_0$, then $(4.6) < 0$, and M has a unique maximum at θ_0. Suppose $\delta = \delta_0$ and $\beta = \beta_0$, and then (4.6) can be written as

$$-P((\delta_0-\beta_0)^T Z)^2[(\mathbf{1}\{\omega^T X \le \gamma\}\mathbf{1}\{\omega_0^T X > \gamma_0\} + \mathbf{1}\{\omega^T X > \gamma\}\mathbf{1}\{\omega_0^T X \le \gamma_0\}]. \quad (4.7)$$

Assuming $\delta_0 \neq \beta_0$, $(4.7) = 0$ when $\mathbf{1}\{\omega^T X \le \gamma\}\mathbf{1}\{\omega_0^T X > \gamma_0\} = 0$ and $\mathbf{1}\{\omega^T X > \gamma\}\mathbf{1}\{\omega_0^T X \le \gamma_0\} = 0$.

Claim 3. *$\mathbf{1}\{\omega^T X \le \gamma\}\mathbf{1}\{\omega_0^T X > \gamma_0\} = \mathbf{1}\{\omega^T X > \gamma\}\mathbf{1}\{\omega_0^T X \le \gamma_0\} = 0$ almost surely if and only if $\omega = \omega_0$ and $\gamma = \gamma_0$.*

Whenever $\omega = \omega_0$ and $\gamma = \gamma_0$, $\mathbf{1}\{\omega^T X \le \gamma\}\mathbf{1}\{\omega_0^T X > \gamma_0\} = \mathbf{1}\{\omega^T X > \gamma\}\mathbf{1}\{\omega_0^T X \le \gamma_0\} = 0$ clearly. Suppose $\mathbf{1}\{\omega^T X \le \gamma\}\mathbf{1}\{\omega_0^T X > \gamma_0\} = \mathbf{1}\{\omega^T X > \gamma\}\mathbf{1}\{\omega_0^T X \le \gamma_0\} = 0$. Define $C_3 = \{(x_1, x_2) : \omega^T X > \gamma, \omega_0^T X \le \gamma_0\}$, and $C_4 = \{(x_1, x_2) : \omega^T X \le \gamma, \omega_0^T X > \gamma_0\}$. Then, the given condition is equivalent to the condition that $P(C_3) = P(C_4) = 0$.

Claim 3.1. If $P(C_3) = P(C_4) = 0$ then $\omega = \omega_0$ and $\gamma = \gamma_0$.

Claim 3.2. IF $\omega \neq \omega_0$ or $\gamma \neq \gamma_0$ then $P(C_3) > 0$ or $P(C_4) > 0$ (equivalent to Claim

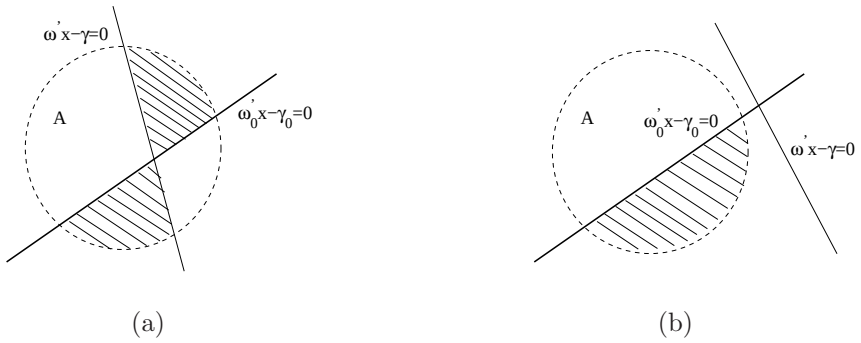
3.1).

Recall assumption **A** in section 4.1 that there exists an open set $A \in \mathbb{R}^2$ such that the density of X on the closure $\bar{A}$ is bounded below and $\omega_0^T X = \gamma_0$ for some $X \in A$. First, suppose that $|\omega_1| \neq |\omega_{0,1}|$ or $|\omega_2| \neq |\omega_{0,2}|$. Since the line $\omega_0^T X - \gamma_0$ passes through A, we have two cases either **a:** the line $\omega^T X - \gamma = 0$ intersects $\omega_0^T X - \gamma_0 = 0$ for some $X \in A$ or **b:** the line $\omega^T X - \gamma = 0$ does not intersect $\omega_0^T X - \gamma_0 = 0$ for any $X \in A$. Figures 4.1(a) and 4.1(b) show the cases of **a** and **b**, respectively. In the case of **a**, there exists an open ball B such that for any $X \in B$, either $P\{\omega^T X - \gamma > 0, \omega_0^T X - \gamma_0 \leq 0\} > 0$ or $P\{\omega^T X - \gamma \leq 0, \omega_0^T X - \gamma_0 > 0\} > 0$, and hence Claim 3.2 is true. In the case of **b**, two lines intersect outside of A, and $C_3 \cup C_4 \cup A \neq \phi$ as shown in Figure 4.1(b). Consequently, whenever $|\omega_1| \neq |\omega_{0,1}|$ or $|\omega_2| \neq |\omega_{0,2}|$, we have $P(C_3) > 0$ or $P(C_4) > 0$, and thus Claim 3.2 is true.

Now, suppose that $|\omega_1| = |\omega_{0,1}|$ and $|\omega_2| = |\omega_{0,2}|$. First, suppose that $\omega_1 = -\omega_{0,1}$, and $\omega_2 = -\omega_{0,2}$. Then $C_3 = \{\omega^T X - \gamma > 0, \omega_0^T X - \gamma_0 \leq 0\} = \{\omega_0^T X - \gamma_0 < -(\gamma + \gamma_0), \omega_0^T X - \gamma_0 \leq 0\}$ is not empty for every $\gamma + \gamma_0 < 0$. Similarly, for every $\gamma + \gamma_0 > 0$, C_4 is not empty. Accordingly, whenever $\omega_1 = -\omega_{0,1}$, and $\omega_2 = -\omega_{0,2}$, we have $P(C_3) > 0$ or $P(C_4) > 0$. If $\omega_1 = \omega_{0,1}$ and $\omega_2 = -\omega_{0,2}$, or if $\omega_1 = -\omega_{0,1}$ and $\omega_2 = \omega_{0,2}$, then the two lines intersect at 90 degrees, and Claim 3.2 is true.

Last, suppose that $\omega_1 = \omega_{0,1}$ and $\omega_2 = \omega_{0,2}$, but $\gamma \neq \gamma_0$. Now, we recall the assumption **B** in the section 4.1. Then,

$$\begin{aligned} P(C_3) = 0 \quad &\Leftrightarrow \quad P\{\omega_0^T X - \gamma_0 > -(\gamma_0 - \gamma), \omega_0^T X - \gamma_0 \leq 0\} = 0 \Rightarrow \gamma - \gamma_0 \geq 0, \\ P(C_4) = 0 \quad &\Leftrightarrow \quad P\{\omega_0^T X - \gamma_0 \leq -(\gamma_0 - \gamma), \omega_0^T X - \gamma_0 > 0\} = 0 \Rightarrow \gamma - \gamma_0 \leq 0. \end{aligned} \quad (4.8)$$



(a) (b)

Figure 4.1: Figures to prove Claim. 3 (a) $\omega^T X - \gamma = 0$ intersects $\omega_0^T X - \gamma_0 = 0$ for some $X \in A$. (b) $\omega^T X - \gamma = 0$ does not intersect $\omega_0^T X - \gamma_0 = 0$ for any $X \in A$. For both cases, there exists an open ball B in which for $X \in B$, $P(\omega^T X - \gamma > 0, \omega_0^T X - \gamma_0 \leq 0) > 0$ or $P(\omega^T X - \gamma \leq 0, \omega_0^T X - \gamma_0 > 0) > 0$.

To satisfy both inequalities of (4.8), γ should be equal to γ_0. In summary, if $\omega_1 \neq \omega_{0,1}$ or $\omega_2 \neq \omega_{0,2}$, then $P(C_3) > 0$ or $P(C_4) > 0$ which verifies Claim 3.2. If $\omega_1 = \omega_{0,1}$ and $\omega_2 = \omega_{0,2}$, then $\gamma = \gamma_0$ in order to satisfy $P(C_3) = P(C_4) = 0$. This verifies Claim 3.1 and, finally, Claim 3. Therefore, assuming $\delta_0 \neq \beta_0$, the equality of the equation (4.6) holds only for $\theta = \theta_0$, and we conclude that M has a unique maximum at θ_0.□

4.2.2 Identifiability conditions

1. $\theta \mapsto M(\theta)$ **is an upper semicontinuous function.**

A function $f : \mathbb{D} \mapsto \mathbb{R}$ is upper semicontinuous if it satisfies either

1. For all $c \in \mathbb{R}$, the set $\{y : f(y) \geq c\}$ is closed or

2. For all $y_0 \in \mathbb{D}$, $\limsup_{y \to y_0} f(y) \leq f(y_0)$.

For all $\theta_0 \in K$, a compact set in H,

$$
\begin{aligned}
M(\theta) &= -P(\epsilon + \beta_0^T Z \mathbf{1}\{\omega_0^T X > \gamma_0\} + \delta_0^T Z \mathbf{1}\{\omega_0^T X \le \gamma_0\} - \beta^T Z \mathbf{1}\{\omega^T X > \gamma\} - \delta^T Z 1\{\omega^T X \le \gamma\})^2 \\
&= -P(\epsilon + (\beta_0 - \delta_0)^T Z \mathbf{1}\{\omega_0^T X > \gamma_0\} + \delta_0^T Z - (\beta - \delta)^T Z \mathbf{1}\{\omega^T X > \gamma\} - \delta^T Z)^2 \\
&= -P\epsilon^2 - 2P\epsilon\{(\beta_0 - \delta_0)^T Z \mathbf{1}\{\omega_0^T X > \gamma_0\} - (\beta - \delta)^T Z \mathbf{1}\{\omega^T X > \gamma\} + (\delta_0 - \delta)^T Z\} \\
&\quad - P((\beta_0 - \delta_0)^T Z \mathbf{1}\{\omega_0^T X > \gamma_0\} - (\beta - \delta)^T Z \mathbf{1}\{\omega^T X > \gamma\} + (\delta_0 - \delta)^T Z)^2 \\
&\le -P\epsilon^2 - 2P\epsilon P(\beta_0 - \delta_0)^T Z \mathbf{1}\{\omega_0^T X > \gamma_0\} + 2P\epsilon P(\beta - \delta)^T Z \mathbf{1}\{\omega^T X > \gamma\} - 2P\epsilon P(\delta - \delta_0)^T Z \\
&\le -P\epsilon^2 = M(\theta_0).
\end{aligned}
$$

Assuming $\epsilon \perp\!\!\!\perp (Z, X)$, $P\epsilon(\beta_0 - \delta_0)^T Z \mathbf{1}\{\omega_0^T X > \gamma_0\} = 0$ because $P\epsilon = 0$, $(\beta_0 - \delta_0)$, and $\mathbf{1}\{\omega_0^T X > \gamma_0\}$ are bounded, and Z is assumed to have a strictly bounded and positive density over a bounded and closed interval. Therefore, $\limsup_{\theta \to \theta_0} M(\theta) \le M(\theta_0)$ for all $\theta_0 \in H$, and by the second definition of the upper semicontinuous functions above, M is an upper semicontinuous function.□

2. $\hat{\theta}_n$ **is uniformly tight.**

We know that a sequence X_n is uniformly tight if X_n is tight for each $n \ge 1$, and it is asymptotically tight. When X_n is a Borel measurable sequence in a Polish space, uniform tightness is equivalent to asymptotic tightness [Van Der Vaart and Wellner, 1996, p.27]. Euclidean space is complete and separable, so hence this space is Polish. Thus it suffices to show that the finite Euclidean estimators $\{\hat{\theta}_n\}$ are asymptotically tight, that is $\|\hat{\theta}_n\| = O_p(1)$. Remember that we restricted $\gamma \in [a, b]$, $\omega \in \mathcal{S}^2$, $\varphi \in H_1$: compact set in $\mathbb{R}^p$. Thus for some constant $K_5 > 0$, we have $\|\hat{\theta}_n\| \le \|\hat{\varphi}_n\| + \|\hat{\omega}_n\| + \|\hat{\gamma}_n\| \le K_5 + 1 + |b - a|$. Now we can show that there exists a positive real number M such that $\lim_{n \to \infty} \|\hat{\theta}_n\| \le M$ taking $K_5 + 1 + |b - a| < M < \infty$, so $\|\hat{\theta}_n\| = O_p(1)$. With $K = [-M, M]$, $\liminf P(\hat{\theta}_n \in K^\delta) \ge 1 - \varepsilon$ for every $\delta > 0$, and

$\varepsilon > 0$. Therefore, by Definition 10 in Chapter 4.1, $\hat{\theta}_n$ is asymptotically tight, so $\hat{\theta}_n$ is uniformly tight.□

So far, we have verified that all conditions in the Argmax theorem are satisfied by M_n, M, $\hat{\theta}$, and θ_0. Therefore $\hat{\theta}_n$ that maximizes the function $M_n(\theta) = P_n m_\theta$ is a consistent estimator for θ_0 in H, where m_θ is defined in (4.2).

4.3 Rate of convergence

In this section, the rates of convergence of the proposed M-estimators of the change-line parameters and regression parameters derive from the limiting behavior of the process $(M_n - M)(\theta)$ following Corollary 1 in Chapter 7.1.

Here is the main results: First, we prove that $M(\theta) - M(\theta_0) \leq -c_1 \tilde{d}^2(\theta, \theta_0)$, where $\tilde{d}^2(\theta, \theta_0) = \|\zeta - \zeta_0\|^2 + \|\omega - \omega_0\| + |\gamma - \gamma_0|$, $c_1 = m_z(k_1 \vee c_2(\|\beta_0 - \delta\|^2 + \|\delta_0 - \beta\|^2))$ for a positive constant m_z which is a lower bound of $P(Z^T Z|\cdot)$, for a positive constant k_1 which is a lower bound of $P(\omega_0^T X > b) \wedge P(\omega_0^T X \leq a)$, and a positive constant c_2. This is true when θ is close to θ_0. $a \vee b$ denotes the maximum between a and b. The second condition is satisfied for $\phi(\eta) = \eta$. The third condition is satisfied for $r_n = \sqrt{n}$. The final inequality is satisfied with $\phi(r_n) = r_n = \sqrt{n}$. Finally, we can prove that $\sqrt{n}|\hat{\zeta}_n - \zeta_0| = O_p(1)$, $n\|\hat{\omega}_n - \omega_0\| = O_p(1)$, and $n|\hat{\gamma}_n - \gamma_0| = O_p(1)$.

lemma 2. *(rates of convergence) Under the regularity conditions given in Chapter 3.1 and 4.1, and consistency provided in Chapter 4.2,* $\sqrt{n}\,|\hat{\zeta}_n - \zeta_0| = O_p(1)$ *and* $n(\|\hat{\omega}_n - \omega_0\| + |\hat{\gamma}_n - \gamma_0|) = O_p(1)$.

4.3.1 $M(\theta) - M(\theta_0) \leq -c_1\tilde{d}^2(\theta, \theta_0)$

$M_\theta - M_{\theta_0}$ can be written as

$$\begin{aligned} M_\theta - M_{\theta_0} &= -P(((\beta_0 - \beta)^T Z)^2 \mathbf{1}\{\omega_0^T X > \gamma_0, \omega^T X > \gamma\}) \\ &- P(((\beta_0 - \delta)^T Z)^2 \mathbf{1}\{\omega_0^T X > \gamma_0, \omega^T X \leq \gamma\}) \\ &- P(((\delta_0 - \beta)^T Z)^2 \mathbf{1}\{\omega_0^T X \leq \gamma_0, \omega^T X > \gamma\}) \\ &- P(((\delta_0 - \delta)^T Z)^2 \mathbf{1}\{\omega_0^T X \leq \gamma_0, \omega^T X \leq \gamma\}). \end{aligned}$$

Note that $P(((\beta_0 - \beta)^T Z)^2 \mathbf{1}\{\omega_0^T X > \gamma_0, \omega^T X > \gamma\}) = P(((\beta_0 - \beta)^T Z)^2 \mid \omega_0^T X > \gamma_0, \omega^T X > \gamma) \times P(\omega_0^T X > \gamma_0, \omega^T X > \gamma)$. Therefore, $M(\theta) - M(\theta_0)$ can be written as follows:

$$\begin{aligned} M_\theta - M_{\theta_0} &= -P(((\beta_0 - \beta)^T Z)^2 \mid \omega_0^T X > \gamma_0, \omega^T X > \gamma) P(\omega_0^T X > \gamma_0, \omega^T X > \gamma) \\ &- P(((\delta_0 - \delta)^T Z)^2 \mid \omega_0^T X \leq \gamma_0, \omega^T X \leq \gamma) P(\omega_0^T X \leq \gamma_0, \omega^T X \leq \gamma) \\ &- P(((\beta_0 - \delta)^T Z)^2 \mid \omega_0^T X > \gamma_0, \omega^T X \leq \gamma) P(\omega_0^T X > \gamma_0, \omega^T X \leq \gamma) \\ &- P(((\delta_0 - \beta)^T Z)^2 \mid \omega_0^T X \leq \gamma_0, \omega^T X > \gamma) P(\omega_0^T X \leq \gamma_0, \omega^T X > \gamma) \\ &= -\|\beta_0 - \beta\|^2 P(Z^T Z \mid \omega_0^T X > \gamma_0, \omega^T X > \gamma) P(\omega_0^T X > \gamma_0, \omega^T X > \gamma) \\ &- \|\delta_0 - \delta\|^2 P(Z^T Z \mid \omega_0^T X \leq \gamma_0, \omega^T X \leq \gamma) P(\omega_0^T X \leq \gamma_0, \omega^T X \leq \gamma) \\ &- \|\beta_0 - \delta\|^2 P(Z^T Z \mid \omega_0^T X > \gamma_0, \omega^T X \leq \gamma) P(\omega_0^T X > \gamma_0, \omega^T X \leq \gamma) \\ &- \|\delta_0 - \beta\|^2 P(Z^T Z \mid \omega_0^T X \leq \gamma_0, \omega^T X > \gamma) P(\omega_0^T X \leq \gamma_0, \omega^T X > \gamma). \end{aligned}$$

We need to assume that $P(Z^T Z \mid \omega_0^T X > \gamma_0, \omega^T X > \gamma)$, $P(Z^T Z \mid \omega_0^T X \leq \gamma_0, \omega^T X \leq \gamma)P$, $P(Z^T Z \mid \omega_0^T X > \gamma_0, \omega^T X \leq \gamma)$, and $P(Z^T Z \mid \omega_0^T X \leq \gamma_0, \omega^T X > \gamma)$ are bounded below by some $m_z > 0$. This is true by the condition that $P(Z^T Z|X)$ is positive definite. Then,

$$
\begin{aligned}
M_\theta - M_{\theta_0} \leq & -\|\beta_0 - \beta\|^2 m_z P_X(\omega_0^T X > \gamma_0, \omega^T X > \gamma) \\
& - \|\delta_0 - \delta\|^2 m_z P_X(\omega_0^T X \leq \gamma_0, \omega^T X \leq \gamma) \\
& - \|\beta_0 - \delta\|^2 m_z P_X(\omega_0^T X > \gamma_0, \omega^T X \leq \gamma) \\
& - \|\delta_0 - \beta\|^2 m_z P_X(\omega_0^T X \leq \gamma_0, \omega^T X > \gamma). \qquad (4.9)
\end{aligned}
$$

We assume that $|(\omega - \omega_0)^T X| < \eta_1$, and $|\gamma - \gamma_0| < \eta_1$ for some $\eta_1 > 0$. This is true when X is in the compact set in $\mathbb{R}^2$ and $\varphi = (\omega, \gamma)$ is in the neighborhood of (ω_0, γ_0). Therefore, we only consider the case where ω is close to ω_0 and γ is close to γ_0. We note that $\gamma_0 - 2\eta_1 < \gamma - (\omega - \omega_0)^T X < \gamma_0 + 2\eta_1$, and $P(\omega_0^T X > b) \wedge P(\omega_0^T X \leq a)$ is bounded below by some constant $k_1 > 0$. Thus

$$
\begin{aligned}
P(\omega_0^T X > \gamma_0, \omega^T X > \gamma) &= P(\omega_0^T X > \gamma_0, \omega_0^T X > \gamma - (\omega - \omega_0)^T X) \\
&\geq P(\omega_0^T X > \gamma_0, \omega_0^T X > \gamma_0 + 2\eta_1) \\
&\geq P(\omega_0^T X > \gamma_0 + 2\eta_1) \\
&> P(\omega_0^T X > b) > k_1
\end{aligned}
$$

by the bounded below condition. Similarly,

$$
\begin{aligned}
P(\omega_0^T X \leq \gamma_0, \omega^T X \leq \gamma) &= P(\omega_0^T X \leq \gamma_0, \omega_0^T X \leq \gamma - (\omega - \omega_0)^T X) \\
&\geq P(\omega_0^T X \leq \gamma_0, \omega_0^T X \leq \gamma_0 - 2\eta_1) \\
&\geq P(\omega_0^T X \leq \gamma_0 - 2\eta_1) \\
&> P(\omega_0^T X < a) > k_1.
\end{aligned}
$$

Therefore, (4.9) satisfies

$$\begin{aligned} M(\theta) - M(\theta_0) &\leq -(\|\beta_0 - \beta\|^2 + \|\delta_0 - \delta\|^2) m_z k_1 \\ &- \|\beta_0 - \delta\|^2 m_z P_X(\omega_0^T X > \gamma_0, \omega^T X \leq \gamma) \\ &- \|\delta_0 - \beta\|^2 m_z P_X(\omega_0^T X \leq \gamma_0, \omega^T X > \gamma). \end{aligned} \tag{4.10}$$

Now we consider $P_X(\omega_0^T X > \gamma_0, \omega^T X \leq \gamma)$ first. Let $E = (\omega - \omega_0)^T X - (\gamma - \gamma_0)$. Then,

$$\begin{aligned} P_X(\omega_0^T X > \gamma_0, \omega^T X \leq \gamma) &= P_X(\omega_0^T X - \gamma_0 > 0, \omega_0^T X - \gamma_0 + (\omega - \omega_0)^T X - (\gamma - \gamma_0) \leq 0) \\ &= P_X(\omega_0^T X - \gamma_0 > 0, \omega_0^T X - \gamma_0 + E \leq 0) \\ &= P_X(\omega_0^T X - \gamma_0 > 0, \omega_0^T X - \gamma_0 + E \leq 0 | E \leq 0) P(E \leq 0) \\ &= P_X(0 < \omega_0^T X - \gamma_0 \leq -E \leq 0 | E \leq 0) P(E \leq 0). \end{aligned} \tag{4.11}$$

This is true because $P_X(\omega_0^T X - \gamma_0 > 0, \omega_0^T X - \gamma_0 + E \leq 0 | E > 0) = 0$. First, we show that $P_X(0 < \omega_0^T X - \gamma_0 \leq -\|\omega - \omega_0\| V + (\gamma - \gamma_0)) \geq c_3 E_V| - \|\omega - \omega_0\| V + \gamma - \gamma_0|$, where $V = \dfrac{(\omega - \omega_0)^T}{\|\omega - \omega_0\|} X$, and for some constant $c_3 > 0$ in Claim 4. Next, we prove that $c_3 E_V| - \|\omega - \omega_0\| V + \gamma - \gamma_0| \geq c_4(\|\omega - \omega_0\| + |\gamma - \gamma_0|)$ for some constant $c_4 > 0$ in Claim 5.

Claim 4. *Let* $V = \dfrac{(\omega - \omega_0)^T}{\|\omega - \omega_0\|} X$. *For some constant* $c_3 > 0$,

$$P_X(0 < \omega_0^T X - \gamma_0 \leq -(\omega - \omega_0)^T X + (\gamma - \gamma_0)) \geq c_3 E_V| - \|\omega - \omega_0\|^T V + (\gamma - \gamma_0)|. \tag{4.12}$$

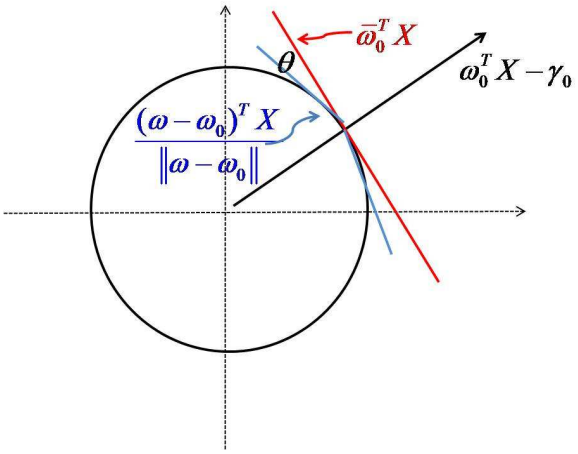


Figure 4.2: The first figure to prove Claim 4. θ is the angle between $\bar{\omega}_0^T X$ and $\frac{(\omega-\omega_0)^T}{\|\omega-\omega_0\|}X$.

Proof. *Again, we only consider the case where* (ω, γ) *are close to* (ω_0, γ_0). *This is enough because we proved consistency in the previous section. We already assumed that* $|(\omega - \omega_0)^T X| < \eta_1$, *and* $|\gamma - \gamma_0| < \eta_1$ *for some* $\eta_1 > 0$. *This means that* $|(\omega - \omega_0)^T X - (\gamma - \gamma_0)| < 2\eta_1$.

Let $u \equiv \omega_0^T X - \gamma_0$, *and* $v \equiv \dfrac{(\omega - \omega_0)^T}{\|\omega - \omega_0\|} X$. *Also, let* $\tilde{v} \equiv \bar{\omega}_0^T X$, *where* $\bar{\omega}_0$ *is an orthogonal vector to* ω_0. *Let* θ *be an angle between* $\bar{\omega}_0$ *and* $(\omega - \omega_0)$. *Since we assumed that* ω *and* ω_0 *are very close,* θ *is very small. Now let* $W = (\omega_0, \bar{\omega}_0)$, *then* $\begin{pmatrix} u \\ \tilde{v} \end{pmatrix} = W^T X$, $X = W \begin{pmatrix} u \\ \tilde{v} \end{pmatrix}$ *and*

$$v = \frac{(\omega - \omega_0)^T W (u,\ \tilde{v})^T}{\|\omega - \omega_0\|} = \frac{(\omega - \omega_0)^T \omega_0}{\|\omega - \omega_0\|} u + \frac{(\omega - \omega_0)^T \bar{\omega}_0}{\|\omega - \omega_0\|} \tilde{v}.$$

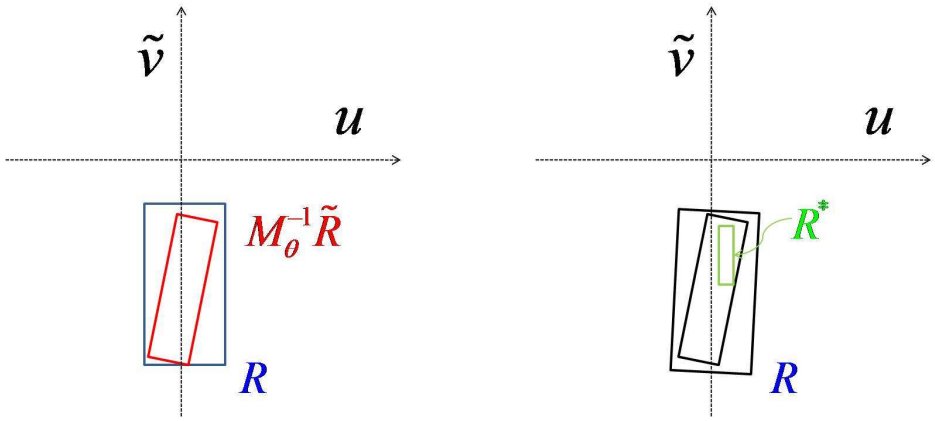
Therefore, (u, v) *can be expressed by*

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ \frac{(\omega-\omega_0)^T\omega_0}{\|\omega-\omega_0\|} & \frac{(\omega-\omega_0)^T\bar{\omega}_0}{\|\omega-\omega_0\|} \end{pmatrix} \begin{pmatrix} u \\ \tilde{v} \end{pmatrix}$$
$$= \begin{pmatrix} 1 & 0 \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} u \\ \tilde{v} \end{pmatrix}$$
$$= M_\theta \begin{pmatrix} u \\ \tilde{v} \end{pmatrix},$$

where $M_\theta = \begin{pmatrix} 1 & 0 \\ \sin\theta & \cos\theta \end{pmatrix}$. *Note that* $< \frac{(\omega-\omega_0)}{\|\omega-\omega_0\|}, \bar{\omega}_0 > = \|\bar{\omega}_0\| \cos\theta = \cos\theta$ *and* $< \frac{\omega-\omega_0}{\|\omega-\omega_0\|}, \omega > = \|\omega_0\| \cos(\frac{\pi}{2} - \theta) = \sin\theta$.

We assume that there exists R *such that for* $(u,\ \tilde{v})^T \in R$, *density* $f(u, \tilde{v}) \geq c$ *for some constant* c. *Then we can find* $0 < \delta \leq 1/2$ *small enough so that there exists* $\tilde{R}$ *with positive density such that* $M_\theta^{-1}\tilde{R} \in R$ *for every* θ *such that* $\cos\theta \geq 1 - \delta$, *where* $\tilde{R}$ *is a shrunk* R. *That is,* $(u,\ \tilde{v}) \in R$, *there exist* $\tilde{R}$ *such that* $(u,\ v) \in \tilde{R} \in M_\theta R$ *as shown in Figure 4.3(a). Note that*

$$\tilde{f}(u, v) = f(M_\theta(u,\ \tilde{v})^T) \left| \begin{matrix} 1 & 0 \\ \sin\theta & \cos\theta \end{matrix} \right| \geq c\cos\theta \geq c(1 - \delta),$$



(a) $M_\theta^{-1}\tilde{R} \in R$ (b) $R^* \in M_\theta^{-1}\tilde{R} \in R$

Figure 4.3: The second and third figures to prove Claim 4.

for all $(u,v) \in \tilde{R}$*, where* $|A|$ *is the determinant of matrix* A*. Therefore,*

$$\begin{aligned} P\mathbf{1}\{0 < \omega_0^T X - \gamma_0 \le -(\omega-\omega_0)^T X + (\gamma-\gamma_0)\} &\ge \int_p \mathbf{1}\{0 < u \le -\|\omega-\omega_0\|v + \gamma - \gamma_0\}dp \\ &= \int_{\tilde{R}} \mathbf{1}\{0 < u \le -\|\omega-\omega_0\|v + \gamma - \gamma_0\}\tilde{f}(u,v)dudv \\ &\ge c(1-\delta)\int_{\tilde{R}} \mathbf{1}\{0 < u \le -\|\omega-\omega_0\|v + \gamma - \gamma_0\}dudv. \end{aligned} \tag{4.13}$$

For the first inequality, for all W *such that* $\cos(\bar{\omega}_0, \omega-\omega_0) \ge 1-\delta$ *for some small* $\delta > 0$*,* R *is a rectangle for* $u, \tilde{v}$*. We consider further shrinkage of* R*. There exists* R^* *with a nontrivial area where* R^* *is as shown in Figure 4.3(b). Then,*

$$\begin{aligned} (4.13) &\ge c(1-\delta)\int_{R^*} \mathbf{1}\{0 < u \le \|\omega-\omega_0\|v + \gamma - \gamma_0\}dudv \\ &\ge \tilde{c}\int_{k_1}^{k_2} (-\|\omega-\omega_0\|v + \gamma - \gamma_0)_+ dv, \end{aligned}$$

where $(A)_+$ *denotes the positive part of* A*. Similarly, we have the following formula*

for another side:

$$P\mathbf{1}\{-(\omega-\omega_0)^T X+\gamma-\gamma_0<\omega_0^T X-\gamma_0\le 0\} \quad \ge \quad \tilde{c}\int_{k_1}^{k_2}(\|\omega-\omega_0\|v+\gamma-\gamma_0)_- dv$$

We can change the constants $\tilde{c}$ *and* $k_1,\ k_2$ *if necessary to make modifications, then the entire form would be as follows:*

$$\ge c^*\int_{k_1}^{k_2}|-\|\omega-\omega_0\|v+\gamma-\gamma_0|dv \tag{4.14}$$

Let V *be uniformly distributed in* $(k_1,\ k_2)$*, then for some constant* $c_3>0$*,*

$$(4.14) \quad \ge \quad c_3 E_v|-\|\omega-\omega_0\|v+\gamma-\gamma_0|,$$

Now, to complete Claim 4, we need to show that $E_V|-\|\omega-\omega_0\|v+\gamma-\gamma_0|$ is bounded below by some positive constant.

Claim 5. *For some positive constant* c*,*

$$E_V|-\|\omega-\omega_0\|^T V+(\gamma-\gamma_0)|\ge c(\|\omega-\omega_0\|+|\gamma-\gamma_0|). \tag{4.15}$$

Claim 5 is verified by the following Theorem:

Theorem 2. *Let* $X\in\mathbb{R}^p$ *be a random vector with* $E(X)=\mu<\infty$*, and suppose there exists a constant* $\eta>0$ *such that* $E|u^T(X-\mu)|\ge\eta$ *for all* $u\in\mathbb{R}^p$ *with* $\|u\|=1$*. Then, there exists a constant* $c>0$ *depending only on* η *and* μ *such that*

$$E|a^T X-b|\ge c(\|a\|+|b|) \tag{4.16}$$

for all $a\in\mathbb{R}^p$ *and* $b\in\mathbb{R}$*.*

Proof. *Theorem 2*

Suppose $\dfrac{|a^T\mu + b|}{\|a\|} \geq \dfrac{\eta}{2}$. *Then,*

$$\begin{aligned} E|a^TX + b| &= \|a\|E\left|\frac{a^T(X-\mu)}{\|a\|} + \frac{a^T(\mu+b)}{\|a\|}\right| \\ &\geq \|a\|\left|E\left(\frac{a^T(X-\mu)}{\|a\|} + \frac{a^T(\mu+b)}{\|a\|}\right)\right| \quad \textit{(by Jensen's Inequality)} \\ &= \|a\|\left|\frac{a^T(\mu+b)}{\|a\|}\right| \\ &\geq \frac{\eta}{2}\|a\|. \end{aligned}$$

Now suppose $\dfrac{|a^T\mu + b|}{\|a\|} < \dfrac{\eta}{2}$. *Then,*

$$\begin{aligned} E|a^TX + b| &= \|a\|E\left|\frac{a^T(X-\mu)}{\|a\|} + \frac{a^T(\mu+b)}{\|a\|}\right| \\ &\geq \|a\|\left(E\left|\frac{a^T(X-\mu)}{\|a\|}\right| - \left|\frac{a^T(\mu+b)}{\|a\|}\right|\right) \\ &\geq \|a\|\left(\eta - \frac{\eta}{2}\right) \\ &\geq \frac{\eta}{2}\|a\|. \end{aligned}$$

Therefore, for all $a \in \mathbb{R}^p$ *and* $b \in \mathbb{R}$,

$$E|a^T(X+b)| \geq \frac{\eta}{2}\|a\|. \tag{4.17}$$

Now suppose $\mu = 0$. *Then*

$$E|a^TX + b| \geq |E(a^TX + b)| = |b| \quad \textit{(by Jensen's Inequality).}$$

Hence, combining with (4.17), we have

$$E|a^T X + b| \geq (\frac{\eta}{2} \wedge 1) \max(\|a\|, b). \tag{4.18}$$

Next, suppose $\mu \neq 0$. *First, assume that* $\|b\| > 2\|\mu\|\|a\|$. *Then,*

$$\begin{aligned} E|a^T X + b| &\geq |b| - |a^T X| \\ &\geq |b| - \|a\|\|X\| \ (|a^T X| \leq \|a\|\|X\|) \\ &\geq \frac{|b|}{2}. \end{aligned}$$

Therefore,

$$E|a^T X + b| \geq \frac{\max(|b|, 2\|\mu\|\|a\|)}{2}. \tag{4.19}$$

Now assume that $\|b\| \leq 2\|\mu\|\|a\|$. *Then, from (4.17), we have*

$$\begin{aligned} E|a^T X + b| &\geq \frac{\eta}{2}\|a\| \\ &= \frac{\eta}{2} \max(\|a\|, \frac{|b|}{2\|\mu\|}) \\ &= \frac{\eta}{2}\frac{1}{2\|\mu\|} \max(|b|, 2\|\mu\|\|a\|). \end{aligned}$$

Combining with (4.19), we have

$$\begin{aligned} E|a^T X + b| &\geq \left(\frac{1}{2} \wedge \frac{\eta}{4\|\mu\|}\right) \max(|b|, 2\|\mu\|\|a\|) \\ &\geq \left(\frac{1}{2} \wedge \frac{\eta}{4\|\mu\|}\right) (1 \wedge 2\|\mu\|) \max(\|a\|, |b|). \end{aligned}$$

Since $\max(\|a\|, |b|) \geq \frac{1}{2}(\|a\| + |b|)$, *we now have, combining everything together, that*

$$E|a^T X + b| \geq c(\|a\| + b),$$

where

$$c = \begin{cases} \frac{1}{2}(\eta/2 \wedge 1), & \text{if } \mu = 0, \\ \frac{1}{2}\left(\frac{1}{2} \wedge \frac{\eta}{4\|\mu\|}\right)(1 \wedge 2\|\mu\|), & \text{if } \mu \neq 0. \square \end{cases}$$

Remark 1. *Under the conditions of the Theorem 2, when* $var(X)$ *exist,* $\exists \eta > 0$ *such that* $\inf_{u \in \mathbb{R}^p,\ \|u\|=1} E|u^T(X-\mu)| \geq \eta$ *if and only if* $var(X)$ *is positive definite.*

Proof. *Remark 1*

Suppose $E|u^T(X-\mu)| \geq \eta > 0\ \forall u : \|u\| = 1$. *Then,*

$$\begin{aligned} u^T var(X) u &= u^T var(X-\mu) u \\ &= var(u^T(X-\mu)) \\ &= E(u^T(X-\mu))^2 - \{E(u^T(X-\mu))\}^2 \\ &= E(|u^T(X-\mu)|^2)\ (\because E(X) = \mu)) \\ &\geq (E|u^T(X-\mu)|)^2\ \textit{(by Jensen's inequality).} \end{aligned}$$

Therefore,

$$\sqrt{u^T var(X) u} \geq E|u^T(X-\mu)| \geq \eta > 0$$

Therefore, $var(X)$ *is positive definite.*

Now suppose that $var(X)$ *is positive definite, that is,* $z^T var(X) z > 0$ *for all* $z > 0,\ z \in \mathbb{R}^p$. *Suppose* $\exists u$ *such that* $\|u\| = 1$ *and* $E|u^T(X-\mu)| = 0$. *Then* $u^T(X-\mu) = 0$ *almost surely, so* $var(X-\mu) = var(X) = 0$. *This makes* $u^T var(X) u = 0$ *which is in contradiction to the positive definiteness of* $var(X)$. *Therefore,* $\exists \eta > 0$ *such that* $E|u^T(X-\mu)| \geq \eta,\ \forall u : \|u\| = 1$. *When* $E|u^T(X-\mu)| = 0$, $u^T(X-\mu) = 0$ *almost surely.*

Suppose $X \in (\mu - \delta_1, \mu + \delta_2)$ *for some positive constants* δ_1, δ_2. *Then,*

$$\begin{aligned} E|u^T(X-\mu)| &= \int_{\mu-\delta_1}^{\mu} -u^T(X-\mu)dx + \int_{\mu}^{\mu+\delta_2} u^T(X-\mu)dx \\ &= \frac{1}{2}u^T((\mu-\delta_1)-\mu)^2 + ((\mu-\delta_2)-\mu)^2 \\ &= 0 \end{aligned}$$

Since $u \neq 0$, $\delta_1 = \delta_2 = 0$. *Therefore,* $u^T(X-\mu) = 0$ *almost surely, and this leads* $Var(u^T(X-\mu) = 0$. $\square$

By combining Claim 4 and Claim 5, we prove that

$$P_X(\omega_0^T X > \gamma_0, \omega^T X \leq \gamma) + P_X(\omega_0^T X \leq \gamma_0, \omega^T X > \gamma) \geq c(\|\omega - \omega_0\| + |\gamma - \gamma_0|), \quad (4.20)$$

where c is a positive constant when $Var(X)$ is positive definite. Now we go back to formula (4.10),

$$\begin{aligned} M(\theta) - M(\theta_0) &\leq -(\|\beta_0 - \beta\|^2 + \|\delta_0 - \delta\|^2)m_z k_1 \\ &- \|\beta_0 - \delta\|^2 m_z P_X(\omega_0^T X > \gamma_0, \omega^T X \leq \gamma) \\ &- \|\delta_0 - \beta\|^2 m_z P_X(\omega_0^T X \leq \gamma_0, \omega^T X > \gamma) \\ &\leq -(\|\beta_0 - \beta\|^2 + \|\delta_0 - \delta\|^2)m_z k_1 \\ &- c_2(\|\omega - \omega_0\| + |\gamma - \gamma_0|)m_z\{\|\beta_0 - \delta\|^2 + \|\delta_0 - \beta\|^2\} \\ &\leq -m_z \max(k_1, c_2(\|\beta_0 - \gamma\|^2 + \|\gamma_0 - \beta\|^2)) \\ &\times (\|\beta - \beta_0\|^2 + \|\delta - \delta_0\|^2 + \|\omega - \omega_0\| + |\gamma - \gamma_0|) \\ &\leq -c_1\tilde{d}^2(\theta, \theta_0), \end{aligned} \quad (4.21)$$

where $c_1 = m_z \max\left(k_1, c_2(\|\beta_0 - \gamma\|^2 + \|\gamma_0 - \beta\|^2)\right) > 0$ and $\tilde{d}^2 = \|\varphi - \varphi_0\|^2 + \|\omega - \omega_0\| + |\gamma - \gamma_0|$, $\varphi = (\beta, \delta)$. Therefore, we have proved that $M(\theta) - M(\theta_0) \lesssim -\tilde{d}^2(\theta - \theta_0)$ for all $\|\theta - \theta_0\|$ small enough. □

4.3.2 Other conditions

1. $E^*\|\mathbb{G}_n\|_{\mathcal{M}_\delta} \lesssim \phi(\delta)$

Again, we can write

$$\begin{aligned} m_\theta - m_{\theta_0} &= -((\delta_0 - \delta)^T Z)^2 \times A_1 - ((\delta - \delta_0)^T Z)^2 \times A_2 \\ &- ((\beta - \delta_0)^T Z)^2 \times A_3 - ((\beta - \beta_0)^T Z)^2 \times A_4 \\ &+ 2\epsilon(\delta_0 - \delta)^T Z \times A_1 + 2\epsilon(\delta - \beta_0)^T Z \times A_2 \\ &+ 2\epsilon(\beta - \delta_0)^T Z \times A_3 + 2\epsilon(\beta - \beta_0)^T Z \times A_4, \end{aligned}$$

where A_1, A_2, A_3 and A_4 were defined in (4.4) in Chapter 4.2. First, we know that $\{\mathbf{1}\{\omega^T X > \gamma\}\}$ is VC, so this class is P-Donsker. We proved that this class is P-Donsker in the proof of consistency. Second, $\{\mathbf{1}\{\gamma < \omega^T X \le \gamma_0\}\}$ is VC, so this class is P-Donsker. The basic idea of the proof is as follows:

1. $\{\gamma < \omega^T X \le \gamma_0\} = \{\omega^T X \le \gamma_0\} \bigcap \{\omega^T X > \gamma\}$

2. $\{\omega^T X - \gamma \ \le 0\}$ is VC with index ≤ 5, and $\{\omega^T X - \gamma_0 \le 0\}$ is VC with index ≤ 5 by Lemma 8 in Chapter 7.1.

3. By Lemma 5 (i) in Chapter 7.1, $\{\omega^T X - \gamma > 0\} = \{\omega^T X - \gamma \le 0\}^c$ is VC with index ≤ 5 again. Also, by Lemma 5 (ii), $\{\omega^T X \le \gamma_0\} \bigcap \{\omega^T X > \gamma\}$ is a VC class of sets with index $\le 5 + 5 - 1$ since both classes are VC with index ≤ 5.

4. by Lemma 6 in Chapter 7.1, $\mathbf{1}\{\gamma < \omega^T X \le \gamma_0\}$ is a VC-subgraph with index ≤ 9 when $\{\gamma < \omega^T X \le \gamma_0\}$ is VC-class of sets with index ≤ 9.

5. Or we can directly use Theorem 4 in Chapter 7.1, so $J(1, \mathcal{F}, L_2) \lesssim 1$. Therefore, $\mathbf{1}\{\gamma < \omega^T X \le \gamma_0\}$ is P-Donsker.

Let $\mathcal{M}_\eta = \{m_\theta - m_{\theta_0} : \tilde{d}(\theta, \theta_0) \le \eta\}$ for small enough η and a discrepancy measure $\tilde{d}$. Then,

$$\mathcal{M}_\eta = \mathcal{M}_\eta^1 + \mathcal{M}_\eta^2 + \mathcal{M}_\eta^3 + \mathcal{M}_\eta^4 + \mathcal{M}_\eta^5 + \mathcal{M}_\eta^6 + \mathcal{M}_\eta^7 + \mathcal{M}_\eta^8,$$

where

$$\begin{aligned}
\mathcal{M}_\eta^1 &= -((\delta_0 - \delta)^T Z)^2 \times A_1, \\
\mathcal{M}_\eta^2 &= -((\delta - \delta_0)^T Z)^2 \times A_2, \\
\mathcal{M}_\eta^3 &= -((\beta - \delta_0)^T Z)^2 \times A_3, \\
\mathcal{M}_\eta^4 &= -((\beta - \beta_0)^T Z)^2 \times A_4, \\
\mathcal{M}_\eta^5 &= 2\epsilon(\delta_0 - \delta)^T Z \times A_1, \\
\mathcal{M}_\eta^6 &= 2\epsilon(\delta - \beta_0)^T Z \times A_2, \\
\mathcal{M}_\eta^7 &= 2\epsilon(\beta - \delta_0)^T Z \times A_3, \\
\mathcal{M}_\eta^8 &= 2\epsilon(\beta - \beta_0)^T Z. \times A_4.
\end{aligned}$$

By Lemma 3 in Chapter 7.1, if $\mathcal{M}_\eta$ is P-Donsker, this is equivalent to that $E^*\|\mathbb{G}_n\|_{\mathcal{M}_\eta} \lesssim \eta$. Since $\mathcal{M}^k, k = 1, 2, \ldots 8$ above are all P-Donsker, $E^*\|\mathbb{G}_n\|_{\mathcal{M}_\eta^k} \lesssim \eta$ for $k = 5, 6, 7, 8$

and $E^*\|\mathbb{G}_n\|_{\mathcal{M}_\eta} \lesssim \eta^2$ for $k = 1, 2, 3, 4$. Therefore,

$$\begin{aligned} E^*\|\mathbb{G}_n\|_{\mathcal{M}_\eta} &\leq E^*\|\mathbb{G}_n\|_{\mathcal{M}_\eta^1} + E^*\|\mathbb{G}_n\|_{\mathcal{M}_\eta^2} \\ &+ E^*\|\mathbb{G}_n\|_{\mathcal{M}_\eta^3} + E^*\|\mathbb{G}_n\|_{\mathcal{M}_\eta^4} \\ &+ E^*\|\mathbb{G}_n\|_{\mathcal{M}_\eta^5} + E^*\|\mathbb{G}_n\|_{\mathcal{M}_\eta^6} \\ &+ E^*\|\mathbb{G}_n\|_{\mathcal{M}_\eta^7} + E^*\|\mathbb{G}_n\|_{\mathcal{M}_\eta^8} \\ &\leq 4\eta + 4\eta^2 \\ &= O(\eta) \end{aligned}$$

Note that $O(\eta^2) = O(\eta)$. Therefore, $E^*\|\mathbb{G}_n\|_{\mathcal{M}_\eta} \leq \eta = \phi(\eta)$. □

2. $M_n(\hat{\theta}_n) \geq \sup_{\theta \in \Theta} M_n(\theta) - O_p(r_n^{-2})$

By the definition of M-estimator in the proof of consistency, we have $M_n(\hat{\theta}_n) \geq \sup_{\theta \in \Theta} M_n(\theta) - o_p(1)$. Therefore,

$$\begin{aligned} \sup_{\theta \in \Theta} M_n(\theta) - O_p(r_n^{-2}) &= \sup_{\theta \in \Theta} M_n(\theta) - O_p(\frac{1}{n}) \\ &\leq \sup_{\theta \in \Theta} M_n(\theta) - o_p(1) \\ &\leq M_n(\hat{\theta}_n). \end{aligned}$$

The condition $M_n(\hat{\theta}_n) \geq \sup_{\theta \in \Theta} M_n(\theta) - O_p(r_n^{-2})$ is satisfied with $r_n = \sqrt{n}$. □

3. $r_n^2 \phi_n(r_n^{-1}) \leq c_3\sqrt{n}$

This is true because $r_n^2\phi(r_n^{-1}) = r_n^2\frac{1}{r_n} = r_n \leq c_3\sqrt{n}$ with $\phi(r_n) = r_n, r_n = \sqrt{n}$. Therefore, by Corollary 1, $\sqrt{n}\tilde{d}(\hat{\theta}_n, \theta_0) = O_p(1)$. We now have $\tilde{d}^2 = \|\hat{\varphi}_n - \varphi_0\|^2 + (|\hat{\gamma}_n - \gamma_0| + \|\hat{\omega}_n - \omega_0\|)$. Therefore, $\sqrt{n}\|\hat{\varphi}_n - \varphi_0\| = O_p(1)$, so the rate of convergence for the model parameter

is $\frac{1}{\sqrt{n}}$. Also, $n|\hat{\gamma}_n - \gamma_0| = O_p(1)$, $n\|\hat{\omega}_n - \omega_0\| = O_p(1)$, so the rate of convergence for the change-line parameters are $\frac{1}{n}$. This result shows that the true change-line parameters can be estimated by a grid search in $\mathcal{S}^2 \times [a, b]$ at the n rate [Pons, 2003]. $\square$

Chapter 5

The interactive decision committee method

5.1 Method

5.1.1 Two-stage cross-validation

As discussed in Hansen and Salamon [1990] and Opitz and Maclin [1999], the decision committee method can reduce test-set error sufficiently by aggregating a few base classifiers, instead of combining all base classifiers. Higher prediction accuracy can be achieved by eliminating some irrelevant or noisy base classifiers. During this selection phase, the forward selection approach is adopted to find optimal combinations of base classifiers similar to Breiman [1996]. In the first step, the best base classifier based on the given prediction accuracy is selected, and denoted by C^1. In the second step, each of the remaining base classifiers $\{C_l^{(2)}\}_{l=1}^{L-1}$ is integrated with C^1 by a given aggregation rule $\mathcal{F}$. The best pair of base classifiers is picked up, and denoted by C^2. For each step of the forward selection approach, the prediction accuracy is assessed, and only one best base classifier is added.

In this study, we propose a 5-fold cross-validation (CV) method to decide the total number of base classifiers K to be included in the final classifier. The training set is randomly split into five subsets, and four out of five subsets are used to train base classifiers. Let $C_{cv,i} = \{C_{cv,i}(z_1), \ldots, C_{cv,i}(z_L)\}$, $i = 1, \ldots, 5$ denote the set of base classifiers for the remaining set which is not used for the ith training. In this phase, we continue the forward selection procedure until all base classifiers are combined. At each step, prediction accuracy is assessed for each of the five sets $\{C_{cv,i}\}_{i=1}^{5}$. Then, we take the average of the prediction accuracies over five sets, and K is decided by the number of combined base classifiers in which the highest average prediction accuracy is achieved. Since another internal 5-fold CV is conducted to determine internal parameters of the SVM learning, a two-stage 5-fold CV is used in this phase. To the best of our knowledge, the proposed two-stage CV is novel.

5.1.2 Univariate and interactive feature space

Suppose that we have the same training data $\{(y_i, x_i)\}_{i=1}^{n}$ as described in section 2.2.2, and testing data $\{x_i\}_{i=n+1}^{I}$. Suppose each feature variable belongs to at least one feature category m, and $\mathbf{x}_m = \{x_{i,j}\}_{i=1,\ j=1}^{n\ \ p_m} \in \mathbb{R}^{n \times p_m}$ denotes a feature matrix for the category m, where p_m is the number of feature variables belonging to the category m, $m = 1, \ldots, M$ and $\sum_{m=1}^{M} p_m = p$. Good examples of these categories would be the blocks of chemical descriptors in chemical toxicity data presented in this study, or e.g., gene ontology terms in gene expression profiles [Ashburner et al., 2000].

Two different feature spaces generated from M feature categories are used to train base classifiers: the univariate feature space and the interactive feature space. The univariate feature space consists of M feature categories of $\mathbf{X} = \{\mathbf{x}_m\}_{m=1}^{M}$. For the interactive feature space, we generate a bivariate feature space $\mathbf{X}^* = \{\mathbf{x}_\mathbf{l}^* = (\mathbf{x_m}, \mathbf{x_{m'}}),$

$m, m' = 1, \ldots, M,\ m \neq m',\ l = 1, \ldots, L = \binom{M}{2}\}$ to construct a new feature space $\mathbf{Z} = \{\mathbf{X} \cup \mathbf{X}^*\}$. By doing this, the interactive feature space allows us to use the information of the feature categories both marginally and interactively. To the best of our knowledge, this study is the first to use the interactive relationship between feature categories to construct base classifiers for decision committees.

5.1.3 UDC and IDC with different aggregation rules

Our proposed method can be summarized in two steps: first, during the construction phase, each base classifier is trained using a feature category $\mathbf{x}_m$ from the univariate feature space or $\mathbf{z}_l$ from the interactive feature space. $C(\mathbf{x}_m)$ and $C(\mathbf{z}_l)$ denote the base classifiers, and $\mu(C(\mathbf{x}_m))$ and $\mu(C(\mathbf{z}_l))$ denote the first-level outputs by using the univariate feature space and the interactive feature space, respectively.

Next, by using two-stage 5-fold CV as described in section 5.1.1, the number of base classifiers K to be aggregated is derived. Once K is decided from the training set, the same forward selection procedures are repeated until we find K base classifiers with the best performance. We call the above decision committee system the Univariate Decision Committee (UDC) if the univariate feature space is used to train the base classifier, and the Interactive Decision Committee (IDC) if both the univariate and the interactive feature spaces are used. The proposed IDC method is new, and we focus on the IDC method in this Chapter.

Figure 5.1 illustrates the basic framework for the IDC method. In this flowchart, X_ms denote the $n \times p_m$ matrix for feature categories $m, m = 1, \ldots, M$, and p_m is the number of variables belonging to feature category m. Z_ls are elements of the interactive feature space, so Z_l is either feature category X_m or a pair of two feature

categories $X_m \cup X_{m'}$, $m \neq m'$ as explained in section 5.1.2. Each base classifier $T(Z_l)$ is trained using feature category Z_l in the training set. The total number of base classifiers K to be combined for the final classifier is determined in this phase by the use of two-stage 5-fold CV. Then, first-level predicted outputs can be obtained from the base classifier $C(Z_l)$s for the test individuals. Finally, the final decision $\hat{C}$ is made by aggregating K base classifiers through using the aggregation rule. In practice, however, we do not have outputs for the new examples. Therefore, we use first-level class predictions from the validation set to find the best K base classifiers to be combined. Then, those selected K base classifiers are combined to predict class labels for new examples.

We utilized voting and stacking to combine base classifiers. For the voting method (denoted by IDC), two aggregation rules $\mathcal{F}_1$ and $\mathcal{F}_2$ as described in section 2.2.2 are utilized to combine base classifiers after the system size K is determined by 5-fold CV. First, we use the aggregation rule of $\mathcal{F}_1$ having $\mu(C(x)) = b(x) = sign(f(x))$ in SVM, $\mu(C(x)) = h_f(x)$ in AdaBoost and random forests as the first-level output. In this study, we set $\omega = 1$ for the unweighted average. The IDC method with this aggregation rule is denoted by $IDC_{\mathcal{F}_1}$. Second, we use the aggregation rule of $\mathcal{F}_2$ having $\mu(C(x)) = f(x)$ as the first-level output, and the IDC method with this rule is denoted by $IDC_{\mathcal{F}_2}$. We apply $\mathcal{F}_2$ for the base classifiers obtained by SVM. In voting, we set the threshold value $c^* = 0$ for the final decision rule. Therefore, $\mathcal{F}_1$ is equivalent to the majority voting method, and $\mathcal{F}_2$ yields the same result as the linear discriminant analysis (LDA) using R as a new feature variable set.

For the IDC method with stacked generalization (denoted by IDC stacking), two different learning algorithms at stage 1 were adopted separately in order to learn

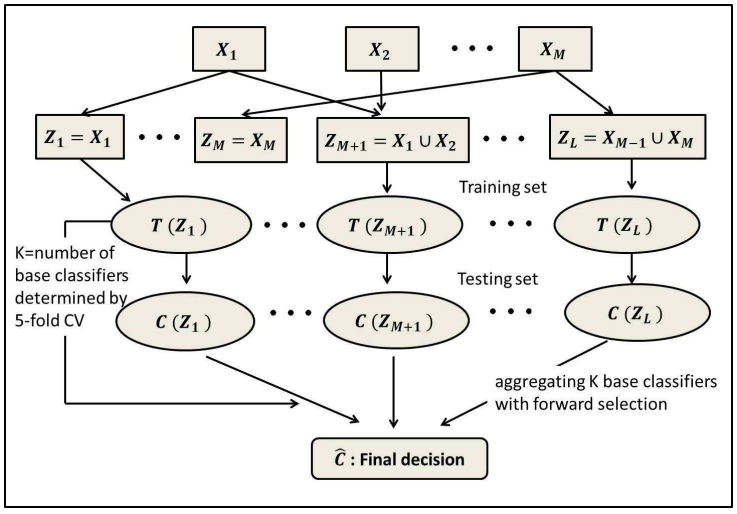


Figure 5.1: Flowchart of the IDC method with M feature categories. $T(Z_l)$ and $C(Z_l)$ denote the base classifiers which are trained using feature set Z_l for the training set and the testing set, respectively.

a combining method: L_2 penalized logistic regression [Park and Hastie, 2008] and a single-hidden layer neural network [Ripley, 2008]. L_2 penalized logistic regression (L_2-logit) was implemented through the **R** package **stepPlr** using BIC (Bayesian Information Criterion) as complexity parameters to compute the score and selecting base classifiers through the forward stepwise (forward select first, then backward deletion follows) selection. A single-hidden layer neural network (NN) was implemented through the **R** package **nnet** with one unit in the hidden layer (single layer), initial random weights on [-0.1, 0.1], a parameter of 0.0005 for weight decay, and a maximum iteration of 300. IDC methods stacked with L_2 penalized logistic regression and NN are denoted by IDC_{LR} and IDC_{NN}, respectively. Note that we do not have to decide on the system size of K for IDC stacking although base classifiers are trained by the same IDC method. Therefore, four different types of aggregation methods are applied to combine base classifiers that are trained by the proposed IDC method.

5.2 Evaluation measure and methods

5.2.1 Prediction accuracy measurement

For the ToxRefDB data, introduced later in this chapter, we have fewer active compounds compared to inactive compounds, which is imbalanced for all binary endpoints. Thus we chose to use both sensitivity and specificity to reflect performance on the classification task [Assareh et al., 2008]. Regarding an active (+1) as positive while an inactive (-1) as negative, sensitivity and specificity are calculated as follows:

$$\begin{aligned} \text{Sensitivity} &= \frac{\text{number of true positives}}{\text{number of true positives} + \text{number of false negatives}}, \\ \text{Specificity} &= \frac{\text{number of true negatives}}{\text{number of true negatives} + \text{number of false positives}}. \end{aligned}$$

Therefore, sensitivity is the proportion of actual active compounds that are correctly classified as active compounds. Similarly, specificity is the proportion of the true inactive compounds which are correctly classified as inactive compounds. The average of sensitivity and specificity was used as a prediction accuracy measure to select base classifiers in forward selection, to decide the number of base classifiers, and to compare the performances on the classification task among different methods:

$$\text{Accuracy} = \frac{\text{sensitivity} + \text{specificity}}{2}.$$

To compare improvement in prediction accuracy relative to the single large, unaggregated classifier, relative improvement (RI) of the classification model M were calculated as follows:

$$\text{RI(M)} = \frac{\text{accuracy of model M} - \text{accuracy of a single large classifier}}{\text{accuracy of a single large classifier}} \times 100.$$

5.2.2 Methods to be compared

First, a single large, unaggregated classifier was used as a reference model (Single). Although random forests and tree-based AdaBoost are already decision committee methods rather than single classifiers, what we really mean by denoting a "single random forests" or a "single AdaBoost" are the usual random forests or AdaBoost without using the IDC method. For a single classifier, we combine the training set and validation set for training. By doing this, we have a larger training set for single classifiers compared to the IDC method and the IDC stacking method.

Second, we apply the proposed IDC method (i.e. determining K and selecting the best K base classifiers by 5-fold CV and forward selection) to each classifier inducer (IDC). We find a training rule and the number of base classifiers to be combined using the training set and decide on the best K base classifiers based on the validation set. Instead of using the training and validation sets separately, one can do another cross-validation by combining the training and validation sets. In our study, using the validation set (a data set that does not contribute to training at all) seems to be slightly better with respect to the performance on the test set than using another cross-validation on the combined training and validation sets (for both training and finding the best K classifiers).

Last, once we trained base-classifiers, we applied stacked generalization with L_2 penalized logistic regression and NN (IDC stacking). IDC and IDC stacking are the same in the first stage, but they combine base classifiers differently.

The primary goal of this study was to compare the interactive decision committee method to non-interactive decision committee methods rather than to find optimal

subsets of features or the optimal classifier inducer. Therefore, comparison between different classification inducers was not done in this study.

5.3 Simulation study

We have empirically evaluated the proposed IDC and IDC stacking methods compared to a single classifier in three classification inducers: SVM, random forests, and tree-based AdaBoost algorithms. Ten random sets of data were generated for the binary classification task, and we used 60% of the data for training, 20% for validation and the remaining set for testing. Again, combined training and validation sets were used for training in a single classifier. For random forests and AdaBoost, 500 and 100 individual trees were grown respectively, with computational costs as a consideration.

5.3.1 Simulation set-up

Four feature categories $X = \{X_i\}_{i=1}^{4}$ were randomly generated, and each category X_i consists of three feature variables $\{X_{ij}\}_{j=1}^{3}$ from the standard multivariate normal distribution $\mathbf{N}_3(0, I)$, where I denotes a 3×3 identity matrix. New variables Zs were generated by combining variables in four feature categories differently so that the effect by univariate or bivariate feature categories could be added to the individual feature variables. Then, we simulated logistic regression models under six different scenarios. A binary outcome was obtained by $Y = \mathbf{1}\{U < p_0\}$, where Us are random variables uniformly distributed in $(0, 1)$, and $p_0 = \dfrac{\exp(\eta)}{1 + \exp(\eta)}$, where η is computed by six different scenarios. Sample size for each run was 300. In AdaBoost, the minimum number of observations that must exist in a node in order for a split to be attempted was 5 and the maximum depth of any node of the final tree was 5 (the root note counted as depth 0). Complexity parameter was C_P, and the weight updating

coefficient was calculated by $\frac{1}{2}\log\frac{1-\text{error}}{\text{error}}$. In random forests, the number of variables randomly sampled at each split was $\sqrt{P}$, where $X \in \mathbb{R}^P$ and the number of minimum observations for the node was 1.

Simulation 1: Two new variables $Z = (Z_1, Z_2)$ were generated, where $Z_1 = (x_{11} + x_{12}) + (x_{21} + x_{22} + x_{23})$, $Z_2 = (x_{31} + x_{32} + x_{33}) + (x_{41} + x4_{42} + x_{43})$. For logistic regression, $\eta = X\beta + Z\gamma$, where $\beta = (\beta_1, \beta_2, \beta_3, \beta_4)^T$, $\beta_1 = (0.12, 0.5, 0.5)^T$, $\beta_2 = (0.15, 0.18, 0.7)^T$, $\beta_3 = (0.11, 0.8, 0.8)^T$, $\beta_4 = (0.5, 0.15, 0.15)^T$, and $\gamma = (1.5, 1.8)^T$.

Simulation 2: Three new variables $Z = (Z_1, Z_2, Z_3)$ were generated, where $Z_1 = (x_{11} + x_{12}) \times (x_{21} + x_{22} + x_{23})$, $Z_2 = (x_{11} + x_{12} + x_{13}) \times (x_{31} + x_{32} + x_{33})$, and $Z_3 = (x_{31} + x_{32} + x_{33}) \times (x_{41} + x_{42} + x_{43})$. For logistic regression, $\eta = X_{12}\beta_{12} + Z\gamma$, where $\beta_{12} = 0.5$ and $\gamma = (1.5, -1.5, 1.8)^T$.

Simulation 3: No new variable for feature categories was derived. $\eta = \beta_{11}X_{11} + \beta_{23}X_{23} + \beta_{31}X_{31} + 1.2X_{11}X_{31}$, where $\beta_{11} = 1.2$, $\beta_{23} = -1.8$, $\beta_{31} = 1.2$.

Simulation 4: No new variable for feature categories was derived. $\eta = X\beta$, where $\beta_1 = (0.12, 0.1, 0.1)^T$, $\beta_2 = (-0.12, -0.1, -0.1)^T$, $\beta_3 = (0.11, 0.1, 0.1)^T$, and $\beta_4 = (-0.11, -0.1, -0.1)^T$.

Simulation 5: Two new variables $Z = (Z_1, Z_2)$ were generated, where $Z_1 = (X_{11} + X_{12}) + (X_{12} + X_{22} + X_{23})$, $Z_2 = (X_{31} + X_{32} + X_{33}) + (X_{41} + X_{42} + X_{43})$. For logistic regression, $\eta = \beta_{12}X_{12} + Z\gamma + 5.0\epsilon$, where $\beta_{12} = 0.5$, $\gamma = (0.1, 0.1)^T$, and random noise $\epsilon \sim \mathbf{N}(0, 1)$.

Simulation 6: Two new variables $Z = (Z_1, Z_2)$ were generated, where $Z_1 = (X_{11} + X_{12}) + (X_{12} + X_{22} + X_{23})$, $Z_2 = (X_{31} + X_{32} + X_{33}) + (X_{41} + X_{42} + X_{43})$. For logistic regression, $\eta = \beta_{12}X_{12} + Z\gamma$, where $\beta_{12} = 0.5$, $\gamma = (0.1, 0.1)^T$.

In Simulation 1, two new variables generated by bivariate linear combination have larger effects compared to individual variables, so Simulation 1 would be favorable to the IDC methods. In Stimulation 2, three new variables were derived by combining feature variables belonging to different feature categories non-linearly, and they have a greater effect than the individual variables. In Simulation 3, three individual variables and one second-order interaction effect by individual variables exist while no intended effect by feature categories exists. In Simulation 4, variables in feature category 1 and variables in feature category 2 have equal effects but with opposite signs. The same is true for the variables in feature category 3 and feature category 4. All variables have small positive effects. In Simulation 5, one weak individual effect and two weak categorical effects exist while a large random error effect exists. Simulation 6 is similar to Simulation 1 except that there is no intended large noise effect.

5.3.2 Main results

Prediction accuracy

Table 5.1 and Figure 5.2 present experimental results under six scenarios. We first focus on the prediction accuracies. In Simulation 1, both the IDC and the IDC stacking methods outperformed single classifiers, especially for the IDC method regardless of classifier inducer (RI: 80.75% for SVM; 57.02% for random forests; 64.88% for AdaBoost). This is not a surprising result since there are greater effects by feature categories.

In Simulation 2, both the IDC and the IDC stacking methods performed similarly to the single classifiers except for the IDC with SVM (18.29%). This indicates that the IDC method might be able to catch a non-linear bivariate structure among feature categories better than a single classifier, but the performance can depend on the

classification inducer.

In Simulation 3, both IDC and IDC staking outperformed single classifiers regardless of classification inducers. IDC (42.53% for SVM; 28.49% for AdaBoost) performed slightly better than IDC stacking (35.92% for SVM; 24.67% for AdaBoost) in SVM and AdaBoost, but the results are comparable to random forests (28.06% for IDC; 28.07% for IDC stacking).

In Simulation 4, both IDC (7.89% for SVM; 8.23% for random forests) and IDC stacking (7.89% for SVM; 9.7% for random forests) performed slightly better than single classifiers in SVM and random forests but similar to or slightly worse than single classifiers in AdaBoost (1.76% for IDC; -0.98% for IDC stacking).

In Simulation 5, single classifiers performed slightly better than IDC (-7.68% for random forests; -9.06% for AdaBoost) and IDC stacking (-4.03% for random forests; -6.79% for AdaBoost) in random forests and AdaBoost, and slightly worse than or similarly to IDC (1.58%) and IDC stacking (-0.59%) in SVM.

In Simulation 6, both IDC (16.57% for SVM; 1.81% for random forests) and IDC stacking (12.97% for SVM; 3.23% for random forests) performed slightly better than single classifiers in SVM and random forests. In AdaBoost, IDC (4.98%) performed slightly better than a single AdaBoost, but slightly worse than IDC stacking (-1.20%).

Based on the empirical results from Simulation 5 and Simulation 6, there appears to be more degradation of performance for the IDC and IDC stacking methods compared to a single classifier, as effects by random noise increased. Simulation 3 shows

that the IDC method can improve prediction accuracy compared to a single classifier when no intended categorical information exists, but a large interaction effect between feature variables belonging to different feature categories exists. Simulation 2 and Simulation 4 show the possibility that IDC may not be able to capture non-linearly associated feature category information or opposite effects between categories well, but it still performs comparatively well compared to single classifiers. Also, IDC and IDC stacking can show different behavior depending on classification inducers and data characteristics.

Standard error estimates in prediction accuracy

In Simulation 1 and Simulation 3, the standard error estimates of the IDC method were smaller than those from single classifiers (overall less than half of the estimates from single classifiers) except for random forests in Simulation 1 (0.025 for single random forests vs. 0.026 for IDC random forests). The standard error estimates of the IDC stacking method were smaller than those from single classifiers, but larger than or similar to those from IDC overall. In Simulation 2, the standard error estimates of the IDC method were slightly smaller than those from single classifiers in SVM and Adaboost, but larger than in random forests (0.017 for single vs. 0.023 for IDC). In Simulation 4, 5, and 6, the standard error estimates of the IDC method were greater than those of single classifiers in SVM, but smaller or similar in the other two classifier inducers.

Overall, the standard error estimates of the IDC methods were smaller than or similar to those of the IDC stacking methods as well as those of single classifiers. The standard error estimates of the IDC stacking with L_2 penalized logistic regression were smaller than or similar to those of the IDC stacking method with NN except

for SVM (0.038 for IDC_{LR}; 0.02 for IDC_{NN}) and AdaBoost (0.021 for IDC_{LR}; 0.011 for IDC_{NN}) in Simulation 6. The empirical results show that the IDC method can perform better than single classifiers with fewer variations when large but relatively simple bivariate feature categorical effects exist or a large interaction effect of individual variables belonging to two feature categories exists. The IDC methods (with voting) perform better than or compare favorably with the IDC stacking method in the current setting.

System size: The number of base classifiers to be combined

For the IDC method, a small investigation was carried out to determine whether the system size of K differed by aggregation methods or by classification inducers. LDA-type aggregation tends to select a smaller number of base classifiers to be combined compared to the unweighted average (majority voting) in SVM. Overall, three classification inducers have a similar committee size K, but Adaboost tends to have smaller number of base classifiers to be combined compared to SVM and random forests except for Simulation 6.

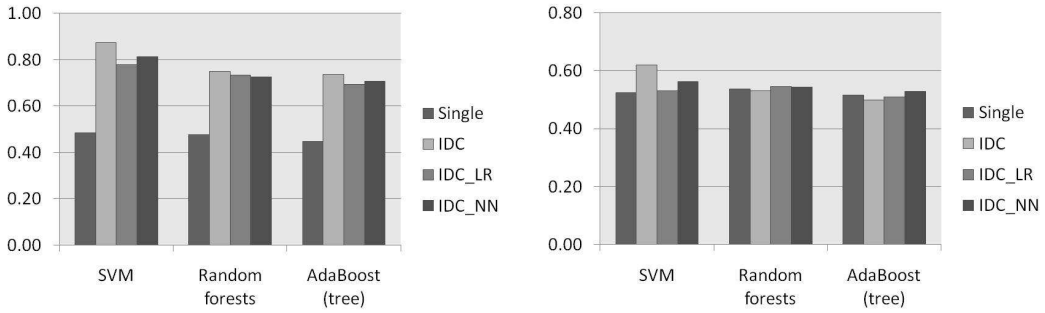
Selected descriptor categories (base classifiers)

Table 5.2 provides the total number of times each descriptor category or pair of descriptor categories were selected in the final classifier over ten replications by the IDC method with SVM, random forests, and tree-based AdaBoost. For SVM, the best results among different kernels and aggregation rules were presented. In Simulation 1, pairs of descriptor categories were selected more frequently than univariate categories ((X2, X4) was selected seven times, (X1, X2) and (X2, X3) were selected six times out of ten replications) by using the IDC method with SVM. In Simulation 3, it is interesting to observe that (X1, X3) was not selected at all for ten replications by

Table 5.1: Averages (ACC) and standard error estimates (SEE) of prediction accuracies over ten replications. Size denotes the number of base classifiers to be combined in the IDC method. For SVM, the best result among three kernel functions are presented indicating which kernel function is the best (l, p, or r). The best method for each classification inducer is marked in **bold**.

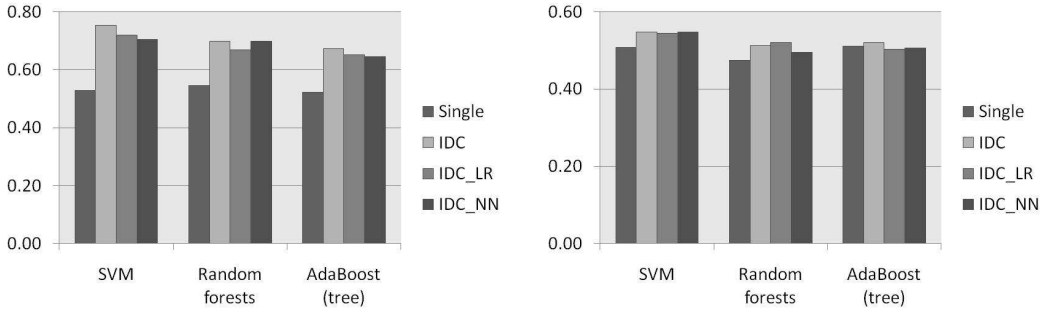
Simulation		SVM					Random forests				AdaBoost (tree)			
		Single	$IDC_{\mathcal{F}_1}$	$IDC_{\mathcal{F}_2}$	IDC_{LR}	IDC_{NN}	Single	IDC	IDC_{LR}	IDC_{NN}	Single	IDC	IDC_{LR}	IDC_{NN}
Sim1	Acc	0.483^r	0.820^p	$\mathbf{0.873}^p$	0.778^p	0.812^p	0.477	**0.749**	0.733	0.725	0.447	**0.737**	0.694	0.707
	SEE	0.035	0.014	0.015	0.023	0.020	0.025	0.026	0.021	0.031	0.026	0.017	0.023	0.025
	Size		6.1	4.0				4.9				4.7		
Sim2	Acc	0.525^p	0.547^r	$\mathbf{0.621}^r$	0.530^r	0.562^r	0.536	0.530	**0.545**	0.544	0.516	0.499	0.509	**0.529**
	SEE	0.027	0.022	0.019	0.021	0.022	0.017	0.023	0.012	0.019	0.017	0.014	0.018	0.028
	Size		3.0	3.8				3.8				3.2		
Sim3	Acc	0.529^p	$\mathbf{0.754}^p$	0.733^p	0.719^p	0.704^p	0.545	0.698	0.668	**0.698**	0.523	**0.672**	0.652	0.646
	SEE	0.036	0.018	0.015	0.017	0.018	0.031	0.012	0.022	0.020	0.029	0.015	0.012	0.022
	Size		4.3	3.7				3.2				4.0		
Sim4	Acc	0.507^l	0.518^r	$\mathbf{0.547}^r$	0.544^r	0.547^p	0.474	0.513	**0.520**	0.495	0.511	**0.520**	0.503	0.506
	SEE	0.015	0.016	0.021	0.016	0.019	0.027	0.015	0.017	0.023	0.019	0.013	0.009	0.028
	Size		3.0	2.7				3.4				2.8		
Sim5	Acc	0.506^l	0.501^l	$\mathbf{0.514}^p$	0.500^p	0.503^l	**0.521**	0.481	0.500	0.493	**0.530**	0.482	0.494	0.489
	SEE	0.013	0.012	0.017	0.010	0.013	0.022	0.018	0.006	0.020	0.028	0.016	0.015	0.016
	Size		2.3	2.0				3.4				2.6		
Sim6	Acc	0.501^p	0.575^p	$\mathbf{0.584}^p$	0.559^p	0.566^p	0.496	0.505	0.508	**0.512**	0.502	**0.527**	0.485	0.496
	SEE	0.025	0.020	0.033	0.038	0.020	0.014	0.021	0.015	0.016	0.022	0.021	0.021	0.011
	Size		3.2	2.4				3.0				3.7		

l=linear, p=quadatic polynomial, and r=radial basis kernel function.
$IDC_{\mathcal{F}_1}$ = IDC = IDC with $\mathcal{F}_1(\cdot|\omega = 1)$
$IDC_{\mathcal{F}_2}$ = IDC with $\mathcal{F}_2(f)$
IDC_{LR} = IDC stacking with L_2 penalized logistic regression.
IDC_{NN} = IDC stacking with NN.



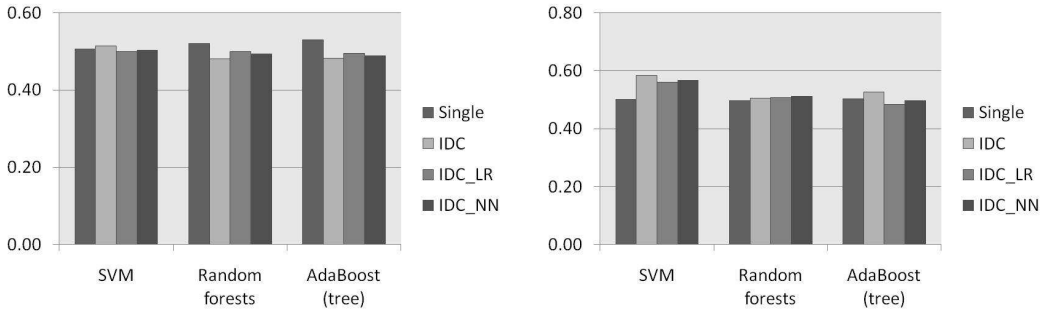
(a) Simulation 1

(b) Simulation 2



(c) Simulation 3

(d) Simulation 4



(e) Simulation 5

(f) Simulation 6

Figure 5.2: Average prediction accuracies over ten replications of single classifier, IDC, and IDC with stacking applied for three classifiers, SVM, random forests, and AdaBoost (tree), are compared. For SVM, the best result among three kernel functions and two aggregation rules are presented. For each classifier inducer, the first bar denotes the prediction accuracy of a single classifier, second bar is for the IDC method, third bar is for the IDC stacking with L_2 logistic regression, and the last bar is for the IDC stacking with NN.

using the IDC method with SVM. Instead, X1 and X3 were selected more frequently than pairs of descriptor categories except for (X1, X2).

By using the IDC method with random forests, (X1, X2) and (X3, X4) pairs were selected seven times and six times, respectively in Simulation 1. Note that we generated two new variables Z1 and Z2 by combining variables belonging to X1 and X2, and variables belonging to X3 and X4, respectively. In Simulation 3, (X1, x2) and (X2, X3) were selected seven times and six times, respectively. This is slightly different from the IDC method with SVM. Instead of selecting univariate categories, pairs of feature categories including significant variables were selected. Note that X4 was selected rarely because X4 alone is not a significant variable.

By using the IDC method with tree-based AdaBoost, similarly to SVM and random forests, pairs of feature categories were selected more frequently than univariate categories ((X2, X3) and (X3, X4) were selected seven times and eight times) in Simulation 1. The (X2, X3) pair was selected the most in Simulation 3 (seven times out of ten replications), and (X1, X2), (X1, X3), and (X2, X4) pairs were selected six times out of ten replications. Again, X4 alone was selected rarely.

5.4 Analysis: Chemical toxicity data

5.4.1 Data description

Chemical toxicity from ToxRefDB

Historical animal toxicity data for 320 compounds are stored in the Toxicity Reference Database (ToxRefDB), developed by the National Center for Computational

Table 5.2: The total number of times each descriptor category or pair of descriptor categories (base classifiers) were selected in the final classifier over ten replications by the IDC method with SVM, random forests, and tree-based AdaBoost. For SVM, the results of the best kernel and aggregation rule are presented.

Category	SVM						random forsts						AdaBoost					
	S1	S2	S3	S4	S5	S6	S1	S2	S3	S4	S5	S6	S1	S2	S3	S4	S5	S6
X1	4	5	6	4	6	4	5	1	1	5	4	4	2	4	3	4	2	7
X2	4	2	3	3	1	4	5	3	4	3	3	2	3	3	3	4	4	3
X3	4	2	9	2	2	1	4	4	2	2	6	0	4	3	5	1	6	0
X4	1	4	1	1	2	2	4	3	1	4	3	5	3	3	2	3	3	5
(X1,X2)	6	5	6	2	1	4	7	7	7	4	7	5	5	6	6	3	2	4
(X1,X3)	3	5	0	0	3	3	4	5	5	2	4	3	4	3	6	2	2	6
(X1,X4)	1	3	0	3	1	1	4	1	0	3	2	4	6	2	0	2	2	8
(X2,X3)	6	4	3	5	1	3	5	1	6	3	3	0	7	5	7	4	3	2
(X2,X4)	7	1	5	4	0	1	5	2	5	7	1	5	5	1	6	3	0	1
(X3,X4)	4	7	5	3	3	1	6	6	1	1	1	2	8	2	2	2	2	1

S1-S6 denote Simulation 1-Simulation 6.

Toxicology in the US Environmental Protection Agency (US EPA) (Martin et al. [2009a]; Martin et al. [2009b]). Up to 78 in-vivo toxicity endpoints are available for each compound. These in-vivo toxicity endpoints were based on chronic, sub-chronic, developmental, and reproductive toxicity experiments. We used a subset of the original data for this study, due to the relatively low ratio of active compounds for most animal toxicity testings. Eighteen endpoints with the highest activity ratios were selected for model development. Also, we excluded duplicates, and those compounds that could not be handled by our descriptor generating software. Across the eighteen endpoints, the number of compounds in each endpoint subset ranged from 237 to 249 (Table 5.3). Toxicity results were coded as 1 (active, toxic), or -1 (inactive, non-toxic).

Chemical toxicity from ICCVAM

A second toxicity data set of 471 compounds was obtained from the Interagency Coordinating Committee on the Validation of Alternative Methods (ICCVAM; NIEHS [2009]). In this data set, the skin sensitization potential was tested using a local lymph

node assay. Each testing result was coded as 1 (active, sensitizer), or -1 (inactive, non-sensitizer). After removing duplicates and the compounds that could not be handled by our descriptor generating software, the final ICCVAM data set contained 262 unique compounds and 134 out of these 262 compounds (51.15%) are active (Table 5.3).

Chemical descriptors computed by DRAGON

For each toxicity data set introduced in section 5.4.1 and 5.4.1, a large set of theoretical molecular descriptors were computed by DRAGON 5.5 software [DRAGON, 2006]. 2,489 chemical descriptors and 2,442 chemical descriptors were available for compounds in the ToxRefDB data and in the ICCVAM data, respectively, after removing descriptors which showed almost no variation in the data set (hereafter "invariant"). The selected chemical descriptors belong to one of the following ten descriptor categories: 2D-autocorrelation (calculated from topological and atomic mass), 1D-functional group counts, 2D-eigenvalue-based indices (all 2D-descriptors based on eigenvalues), 2D-molecular properties (measures of certain physical properties), 2D-atom-centered fragments, 2D-topological descriptors (a number of topological patterns), 2D-connectivity indices (number of indices), 0D-constitutional descriptors (number of atoms), 2D-walk and path counts, and 2D-fingerprints. These categories are different logical blocks of molecular descriptors computed by DRAGON. For each data set, the ten categories of the chemical descriptors and the number of chemical descriptors belonging to each category after removing invariant descriptors are given in Table 5.4. Ten different feature categories are associated with theoretical molecular structure, it is reasonable to build base classifiers based on the various feature categories and to combine base classifiers using a decision committee method.

Table 5.3: All endpoints for the chemical toxicity and the total number of available chemical compounds for each endpoint are given for both chemical toxicity data sets. The numbers in parentheses denote the percentage of active compounds for each endpoint.

Data	Endpoints	Toxicity category	Test species	Total number of compounds (% of active compounds)
ToxRefDB	CHR: Mouse: Liver Hypertrophy (Y1)	chronic	mouse	239 (27.62 %)
	CHR: Mouse: Liver Proliferative Lesions (Y2)	chronic	mouse	239 (38.91 %)
	CHR: Mouse: Liver Tumors (Y3)	chronic	mouse	239 (30.13 %)
	CHR: Mouse: Tumorigen (Y4)	chronic	mouse	239 (38.49 %)
	CHR: Rat: Liver Hypertrophy (Y5)	chronic	rat	247 (26.32 %)
	CHR: Rat: Liver Proliferative Lesions (Y6)	chronic	rat	247 (26.32 %)
	CHR: Rat: Tumorigen (Y7)	chronic	rat	247 (39.27 %)
	DEV: Rabbit: General Fetal Weight Reduction (Y8)	developmental	rabbit	237 (20.68 %)
	DEV: Rabbit: Pregnancy Related Embryo Fetal Loss (Y9)	developmental	rabbit	237 (29.54 %)
	DEV: Rabbit: Pregnancy Related Maternal Preg Loss (Y10)	developmental	rabbit	237 (45.99 %)
	DEV: Rabbit Skeletal Axial (Y11)	developmental	rabbit	237 (23.21 %)
	DEV: Rat: General Fetal Weight Reduction (Y12)	developmental	rat	249 (34.94 %)
	DEV: Rat: Pregnancy Related Embryo Fetal Loss (Y13)	developmental	rat	249 (22.09 %)
	DEV: Rat: Pregnancy Related Maternal Preg Loss (Y14)	developmental	rat	249 (19.68 %)
	DEV: Rat: Skeletal Axial (Y15)	developmental	rat	249 (44.58 %)
	MGR:Rat: Kidney (Y16)	reproductive	rat	244 (30.33 %)
	MGR: Rat: Liver (Y17)	reproductive	rat	244 (42.62 %)
	MGR: Rat: Viability PND4 (Y18)	reproductive	rat	244 (27.87 %)
ICCVAM	Skin sensitization			262 (51.15 %)

Table 5.4: Ten categories of chemical descriptors. The number of descriptors belonging to each category was obtained after removing invariant descriptors. These molecular descriptors were derived using DRAGON 5.5 software.

Category of variables	Number of variables	
	ToxRefDB	ICCVAM
0D-constitutional descriptors	40	46
2D-topological descriptors	98	102
2D-walk and path counts	47	47
2D-connectivity indices	33	33
2D-autocorrelations	96	96
2D-eigenvalue-based indices	235	235
1D-functional group counts	64	85
2D-atom-centered fragments	81	79
2D-molecular properties	28	29
2D-fingerprints	382	378
Total	1,104	1,130

5.4.2 Main results

In this section, we describe the empirical results of applying the IDC methods and the IDC stacking method as well as a single classifier to two chemical toxicity data sets. Again, three classifier inducers were explored: SVM, random forests, and tree-based AdaBoost. For stacked generalization, L_2 penalized logistic regression with stepwise selection and a single layer NN were adopted. Ten replications were obtained randomly. In each run, the data set was randomly split into three sets, and we used 60% of the data for training, 20% for validation, and the remaining set for testing. The average prediction accuracies of the ten replications were compared. Again, 500 and 100 individual trees were grown in random forests and AdaBoost respectively. In SVM, linear and radial basis functions were utilized without optimizing any other parameters, considering computational cost.

Prediction accuracy

1. ToxRefDB data set. Figure 5.3 and Table 5.5 display the average of the prediction accuracies computed by using the test set. In SVM, the highest accuracies between linear and radial basis kernel are presented. For the IDC method, the best prediction accuracies between $IDC_{\mathcal{F}_1}$ and $IDC_{\mathcal{F}_2}$ are reported. In SVM, the prediction accuracy of the IDC method achieved the highest prediction accuracies for 14 endpoints (Y1, Y2, Y3, Y5, Y6, Y8, Y9, Y10, Y11, Y12, Y13, Y14, Y16, Y17), especially for Y10 (RI: 7.24%), Y13 (8.2%), and Y17 (7.26%). IDC stacking performed best for three endpoints (Y4 and Y18 for IDC stacking with L_2-logit and Y7 for IDC stacking with NN). In Y15, a single SVM achieved the highest prediction accuracy.

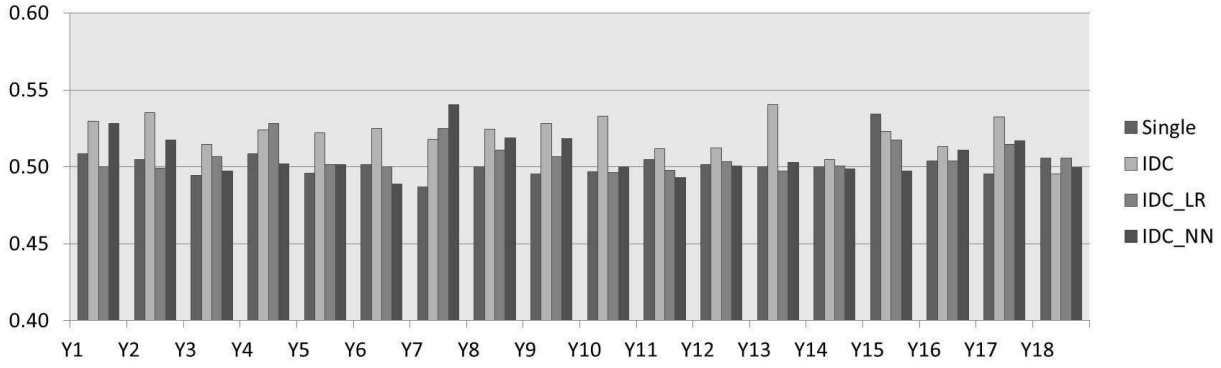
With random forests, the IDC method achieved the highest prediction accuracies for 7 endpoints (Y1, Y8, Y9, Y10, Y13, Y15, and Y17), especially for Y8 (RI: 9.96%), Y10 (16.36%) and Y15 (7.62%). IDC stacking performed better than the IDC methods as well as single classifiers for 7 endpoints (Y7, Y11, and Y14 for IDC stacking with L_2-logit and Y3, Y4, Y5, and Y12 for IDC stacking with NN).

Using Adaboost, the IDC method performed best for 8 endpoints (Y1, Y5, Y8, Y9, Y10, Y12, Y15, and Y17), especially for Y10 (RI: 9.88%). IDC stacking achieved the highest prediction accuracies for 9 endpoints (Y3, Y7, Y11 for IDC stacking with L_2-logit and Y2, Y4, Y13, Y14, Y16, and Y18 for IDC stacking with NN). A single AdaBoost tree performed best for Y6.

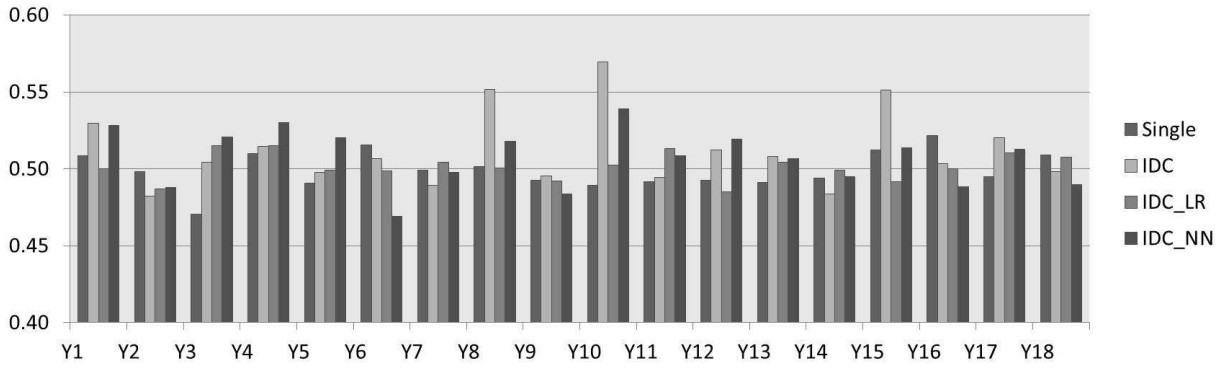
The empirical results show that the IDC method was the best choice for 10 endpoints: Y1 ($IDC_{\mathcal{F}_2}$ applying SVM with linear kernel), Y5($IDC_{\mathcal{F}_1}$ applying SVM with linear kernel), Y6 ($IDC_{\mathcal{F}_1}$ applying SVM with RBF kernel), Y8 (IDC applying

random forests), Y9 ($IDC_{\mathcal{F}_1}$ applying SVM with RBF kernel), Y10 (IDC applying random forests), Y13 ($IDC_{\mathcal{F}_2}$ applying VM with RBF kernel), Y14 ($IDC_{\mathcal{F}_1}$ applying SVM with linear kernel), Y15 (IDC applying random forests), and Y17 ($IDC_{\mathcal{F}_1}$ applying SVM with linear kernel). IDC stacking was the best setting for 6 endpoints: Y2 (IDC stacking with L_2 penalized logistic applying AdaBoost tree), Y3, Y4 (IDC stacking with NN applying random forests), Y7 (IDC stacking with NN applying SVM with linear kernel), Y11 (IDC stacking with L_2 penalized logistic applying random forests), and Y12 (IDC stacking with NN applying random forests). Both IDC and IDC stacking methods failed to improve prediction accuracies compared to a single classifier for Y16 and Y18, and single random forests achieved the best performance in the current experimental setting. Overall, the classification performance was not very good, and the IDC or the IDC stacking methods are not always better than a single classifier. The experimental results, however, show that the IDC method and the IDC stacking method perform as well or better than single classifiers for the majority of endpoints in the ToxRefDB data, especially applying the SVM method which is kernel based, and base classifiers are not trained by a decision committee method.

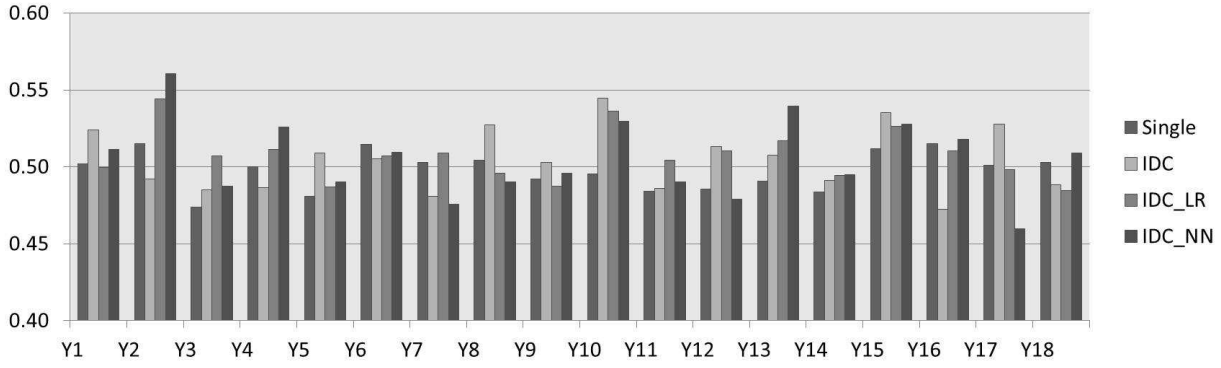
2. ICCVAM data set. Figure 5.4 and Table 5.5 provide the average of prediction accuracies for ICCVAM data set. For the ICCVAM data, all examined methods produced better prediction accuracies compared to those in the ToxRefDB data. The IDC method achieved the highest prediction accuracy applying SVM (RI: 6.07% with linear kernel) and the AdaBoost tree (1.46%), but failed to improve prediction accuracy applying random forests (-3.78%). Overall, we could not find a substantial difference among the explored methods, but single random forests achieved the highest performance in the current setting.



(a) SVM



(b) Random forests



(c) AdaBoost

Figure 5.3: ToxRefDB data set: (a) Prediction accuracy (accuracy of the best model among different kernels and aggregation rules) in SVM (b) Prediction accuracy in random forests (c) Prediction accuracy in AdaBoost for each of eighteen endpoints (e.g. Y1 is CHR: Mouse: Liver Hypertrophy from the Table 5.3).

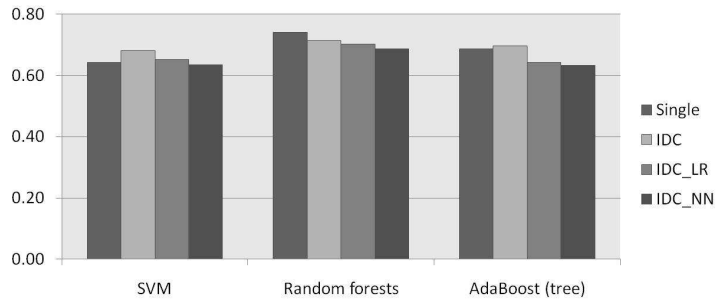


Figure 5.4: ICCVAM data set: Prediction accuracy of the best model among different kernels for SVM.

System size: The number of base classifiers to be combined

Opitz and Maclin [1999] investigated the appropriate number of base classifiers for the final classifier in bagging and boosting. The authors reported that much of the reduction in test-set error was observed at 10 to 15 base classifiers in bagging and boosting applied to neural networks and in bagging applied to decision trees. Adaboost applied to a decision tree continued to reduce test-set error until 25 base classifiers were aggregated.

1. ToxRefDB data set. Table 5.6 provides the mean and standard error estimates of the number of base classifiers to be aggregated by the IDC method over ten replications, which was decided by two-stage 5-fold CV. Applying SVM, $IDC_{\mathcal{F}_2}$ tends to have more base classifiers (on average, across all 18 endpoints, 15.27 base classifiers) than the unweighted average $IDC_{\mathcal{F}_1}$ (on average, 6.07) with greater variation, especially for the radial basis kernel (on average, across 18 endpoints, standard error estimates were 0.702 and 2.153 for $IDC_{\mathcal{F}_1}$ and $IDC_{\mathcal{F}_2}$, respectively). However, the number of base classifiers was still less than half of all base classifiers for all four

Table 5.5: Averages (ACC) and standard error estimates(SEE) of prediction accuracies over ten replications for ToxRefDB and ICCVAM data. For SVM, the best results among three kernel functions are presented indicating which kernel function is the best [l=linear and r=radial basis function]. The best method for each classification inducer is marked in **bold**.

Data	Endpoints		SVM					Random forests				AdaBoost (tree)			
			Single	$IDC_{\mathcal{F}_1}$	$IDC_{\mathcal{F}_2}$	IDC_{LR}	IDC_{NN}	Single	IDC	IDC_{LR}	IDC_{NN}	Single	IDC	IDC_{LR}	IDC_{NN}
ToxRefDB	Y1	ACC	0.509^r	0.518^l	$\mathbf{0.530}^l$	0.500^r	0.528^l	0.489	0.480	0.498	**0.515**	0.502	**0.524**	0.499	0.511
		SEE	0.009	0.022	0.026	0.000	0.020	0.011	0.019	0.008	0.022	0.015	0.014	0.015	0.019
	Y2	ACC	0.505^r	$\mathbf{0.535}^l$	0.508^r	0.499^r	0.517^l	**0.498**	0.482	0.487	0.488	0.515	0.492	0.544	**0.561**
		SEE	0.020	0.029	0.021	0.016	0.021	0.023	0.019	0.020	0.023	0.022	0.023	0.017	0.028
	Y3	ACC	0.494^r	$\mathbf{0.515}^r$	0.491^l	0.507^r	0.497^r	0.471	0.504	0.515	**0.520**	0.474	0.485	**0.507**	0.488
		SEE	0.003	0.029	0.031	0.007	0.015	0.011	0.019	0.020	0.018	0.018	0.017	0.028	0.023
	Y4	ACC	0.509^r	0.488^l	0.524^r	$\mathbf{0.528}^l$	0.502^l	0.510	0.515	0.515	**0.530**	0.500	0.486	0.511	**0.526**
		SEE	0.016	0.013	0.022	0.026	0.029	0.017	0.014	0.020	0.026	0.013	0.028	0.017	0.026
	Y5	ACC	0.496^r	$\mathbf{0.522}^l$	0.520^l	0.502^r	0.502^l	0.491	0.498	0.499	**0.520**	0.481	**0.509**	0.487	0.490
		SEE	0.002	0.026	0.026	0.003	0.010	0.010	0.016	0.012	0.020	0.011	0.019	0.012	0.022
	Y6	ACC	0.501^r	$\mathbf{0.525}^r$	0.500^r	0.500^r	0.489^r	**0.515**	0.507	0.499	0.469	**0.515**	0.505	0.507	0.509
		SEE	0.003	0.026	0.027	0.000	0.006	0.010	0.024	0.005	0.016	0.010	0.021	0.015	0.021
	Y7	ACC	0.487^r	0.518^r	0.498^l	0.525^r	$\mathbf{0.541}^l$	0.499	0.489	**0.504**	0.498	0.503	0.481	**0.509**	0.476
		SEE	0.009	0.022	0.025	0.012	0.018	0.018	0.019	0.023	0.019	0.015	0.021	0.020	0.023
	Y8	ACC	0.500^r	0.506^r	$\mathbf{0.525}^l$	0.511^r	0.519^r	0.502	**0.552**	0.501	0.518	0.505	**0.528**	0.496	0.490
		SEE	0.000	0.022	0.023	0.008	0.009	0.009	0.022	0.015	0.020	0.010	0.021	0.011	0.019
	Y9	ACC	0.495^r	$\mathbf{0.528}^r$	0.510^l	0.507^l	0.518^l	0.493	**0.495**	0.492	0.484	0.492	**0.503**	0.487	0.496
		SEE	0.002	0.034	0.023	0.014	0.023	0.013	0.023	0.008	0.015	0.015	0.016	0.016	0.020
	Y10	ACC	0.497^r	0.515^r	$\mathbf{0.533}^l$	0.496^l	0.500^r	0.489	**0.569**	0.502	0.539	0.496	**0.545**	0.536	0.530
		SEE	0.018	0.011	0.026	0.027	0.036	0.022	0.016	0.019	0.022	0.022	0.014	0.021	0.020
	Y11	ACC	0.505^r	$\mathbf{0.512}^l$	0.506^l	0.498^r	0.493^r	0.491	0.494	**0.513**	0.508	0.484	0.486	**0.505**	0.490
		SEE	0.005	0.011	0.019	0.003	0.004	0.011	0.018	0.016	0.018	0.010	0.023	0.013	0.019
	Y12	ACC	0.502^r	$\mathbf{0.512}^l$	0.496^r	0.503^r	0.501^r	0.493	0.512	0.485	**0.519**	0.485	**0.513**	0.510	0.479
		SEE	0.006	0.017	0.022	0.014	0.018	0.022	0.021	0.015	0.026	0.019	0.020	0.022	0.021
	Y13	ACC	0.500^r	0.504^r	$\mathbf{0.541}^r$	0.497^r	0.503^r	0.491	**0.508**	0.504	0.506	0.491	0.507	0.517	**0.539**
		SEE	0.000	0.031	0.030	0.003	0.005	0.007	0.004	0.007	0.023	0.010	0.019	0.021	0.020
	Y14	ACC	0.500^r	$\mathbf{0.505}^l$	0.503^l	0.500^1	$\mathbf{0.499}^2$	0.494	0.483	0.499	0.495	0.484	0.491	0.494	**0.495**
		SEE	0.005	0.008	0.022	0.016	0.001	0.011	0.017	0.014	0.011	0.006	0.016	0.010	0.015
	Y15	ACC	$\mathbf{0.534}^r$	0.524^l	0.511^r	0.517^l	0.497^r	0.512	**0.551**	0.492	0.513	0.512	**0.535**	0.526	0.528
		SEE	0.021	0.025	0.025	0.021	0.027	0.031	0.023	0.026	0.015	0.021	0.020	0.017	0.016
	Y16	ACC	0.504^r	$\mathbf{0.513}^r$	0.480^r	0.504^r	0.511^r	**0.522**	0.503	0.500	0.489	0.515	0.472	0.510	**0.518**
		SEE	0.004	0.010	0.031	0.003	0.008	0.019	0.018	0.010	0.017	0.011	0.014	0.019	0.018
	Y17	ACC	0.496^r	$\mathbf{0.532}^l$	0.514^l	0.515^r	0.517^l	0.495	**0.520**	0.510	0.513	0.501	**0.528**	0.498	0.460
		SEE	0.012	0.019	0.017	0.015	0.017	0.022	0.021	0.022	0.021	0.017	0.020	0.024	0.029
	Y18	ACC	0.506^r	0.495^l	0.484^l	$\mathbf{0.506}^l$	0.500^l	**0.509**	0.498	0.508	0.490	0.503	0.488	0.485	**0.509**
		SEE	0.006	0.013	0.021	0.024	0.026	0.014	0.024	0.015	0.018	0.013	0.027	0.015	0.022
ICCVAM		ACC	0.642^l	$\mathbf{0.681}^r$	0.646^r	0.651^r	0.634^r	**0.741**	0.713	0.701	0.687	0.687	**0.697**	0.642	0.632
		SEE	0.018	0.022	0.028	0.021	0.017	0.016	0.017	0.024	0.012	0.012	0.016	0.029	0.020

models. The IDC method applying random forests and AdaBoost tree tend to combine a smaller number of base classifiers compared to the IDC method applying SVM (on average, across 18 endpoints, 3.16 and 3.68 base classifiers for random forests and AdaBoost tree, respectively) with less variation (on average, standard error estimates were 0.331 and 0.404 for random forests and AdaBoost tree, respectively).

Since the number of base classifiers for the final classifier was determined by 5-fold CV, we explored variation in the total number of base classifiers within 5-fold CV. This investigation was limited to the SVM classifier. Table 5.7 provides the average and standard error estimates of the standard deviation of the number of base classifiers within 5-fold CV over ten replications. The standard deviation was computed by using the fifth part of the training set which was not used for training. The smaller variation within CV was observed when the unweighted average aggregation rule $IDC_{\mathcal{F}_1}$ was applied (average standard deviation was 4.00 and 1.01 for the linear and the radial basis kernels, respectively) compared to the LDA-type aggregation rule $IDC_{\mathcal{F}_2}$ (average standard deviation was 9.56 and 10.87 for the linear and the radial basis kernels, respectively). In this study, 5-fold CV was selected considering the small sample size and computational cost. This result indicates that the system size K determined by the 5-fold CV has a relatively large variation, and it would be better to apply 10-fold CV to reduce variation for larger data sets.

2. ICCVAM data set. Similar to the results from the ToxRefDB data set, $IDC_{\mathcal{F}_2}$ applying SVM tends to combine a larger number of base classifiers as compared to $IDC_{\mathcal{F}_1}$ applying SVM (on average, 8.3 and 17.4 for $IDC_{\mathcal{F}_1}$ and $IDC_{\mathcal{F}_2}$, respectively) with greater variation (on average, 0.633 and 1.881 for $IDC_{\mathcal{F}_1}$ and $IDC_{\mathcal{F}_2}$, respectively). Unlike the ToxRefDB data set, IDC with aggregation by the unweighted

average applying random forests and AdaBoost tree produced a system size similar to the IDC with the same aggregation rule applying SVM (on average, 8.3 and 8.8 for random forests and AdaBoost, respectively) with greater variation (0.907 and 0.998 for random forests and AdaBoost tree, respectively).

Averages and standard error estimates of the standard deviation of the number of base classifiers within 5-fold CV are given in Table 5.7. The difference between different aggregation methods was slightly smaller than those in the ToxRefDB, but the unweighted average aggregation rule $IDC_{\mathcal{F}_1}$ (average standard deviation was 6.07 and 3.87 for the linear and the radial basis kernels, respectively) produced less variation than the LDA-type aggregation rule $IDC_{\mathcal{F}_2}$ (average standard deviation was 8.49 and 7.51 for the linear and the radial basis kernels, respectively) which is similar to the results in the ToxRefDB data set.

Selected descriptor categories (base classifiers)

In this study, we focused on improvement on the classification task rather than the feature selection problem, although diversity among base classifiers was increased through using heterogeneous subsets of feature variables. However, it is still be worthwhile to see which descriptor category was most frequently selected.

1. ToxRefDB data set. Tables 5.8, 5.9, and 5.10 provide the total number of times each descriptor category or pairs of descriptor categories were selected in the final classifier over ten replications for the IDC method with SVM, random forests, and tree AdaBoost, respectively. With SVM, 1D-functional group and 2D-molecular

Table 5.6: Averages (AVG) and standard error estimates (SEE) of the number of base classifiers to be combined which is determined by 5-fold CV over ten replications for ToxRefDB and ICCVAM data. For SVM, the best results among linear and radial basis function kernels are presented.

Data	Endpoints	SVM				Random forests		AdaBoost (tree)	
		$IDC_{\mathcal{F}_1}$		$IDC_{\mathcal{F}_2}$					
		AVG	SEE	AVG	SEE	AVG	SEE	AVG	SEE
ToxRefDB	Y1	9.4	0.933	14.5	2.377	2.8	0.442	2.6	0.427
	Y2	9.7	1.075	14.4	2.212	5	0.683	4.6	0.306
	Y3	8.3	0.367	13.2	1.569	3.2	0.533	2.8	0.327
	Y4	3.2	0.800	17.7	2.556	3.4	0.521	4.8	0.533
	Y5	6.2	0.533	13.2	1.919	2	0.000	2.6	0.427
	Y6	7.7	1.096	18.7	2.246	2.4	0.267	2.4	0.267
	Y7	3	0.333	11.8	2.215	4.9	0.482	5.6	1.327
	Y8	9.3	1.033	17.6	1.500	1.9	0.100	2	0.000
	Y9	7.3	0.667	9.9	1.581	2.2	0.200	2.2	0.200
	Y10	11.1	1.169	19.2	2.764	5.2	0.533	8	0.667
	Y11	1.4	0.163	13.7	1.627	2.4	0.267	2	0.000
	Y12	9.2	0.964	14.6	1.675	3.2	0.533	4.4	0.581
	Y13	10.1	2.163	18.3	3.461	2	0.000	2	0.000
	Y14	1.1	0.100	16.6	3.557	2	0.000	2	0.000
	Y15	5.8	1.052	13.9	1.748	5.2	0.442	6.8	0.854
	Y16	1.6	0.163	17	2.186	2.2	0.200	2.4	0.267
	Y17	2.8	0.442	14.8	1.692	4	0.422	6.2	0.757
	Y18	2	0.000	15.7	1.862	2.8	0.327	2.8	0.327
ICCVAM		8.3	0.633	17.4	1.881	8.3	0.907	8.8	0.998

Table 5.7: Averages and standard error estimates of the standard deviation in the total number of base classifiers within 5-fold CV in IDC applying SVM for the ToxRefDB and the ICCVAM data sets.

Data	Endpoints	$IDC_{\mathcal{F}_1}$				$IDC_{\mathcal{F}_2}$			
		Linear		RBF		Linear		RBF	
		AVG	SEE	AVG	SEE	AVG	SEE	AVG	SEE
ToxRefDB	Y1	3.30	0.49	1.80	1.27	10.03	1.28	12.05	1.63
	Y2	3.99	0.40	1.08	0.17	9.94	1.37	9.51	1.26
	Y3	3.97	0.85	0.93	0.30	6.77	0.86	11.31	1.20
	Y4	5.02	0.47	1.51	0.50	11.32	1.62	11.30	1.82
	Y5	4.15	0.31	0.36	0.14	9.07	1.38	11.99	1.21
	Y6	3.14	0.29	0.30	0.08	8.68	1.64	11.14	1.37
	Y7	4.22	0.62	1.25	0.14	7.83	1.06	10.28	1.06
	Y8	3.85	0.56	0.09	0.06	10.13	1.44	9.88	2.11
	Y9	3.97	0.38	0.48	0.07	9.87	1.65	9.80	1.25
	Y10	4.06	0.76	2.87	0.44	10.96	1.11	9.98	1.28
	Y11	5.71	0.56	0.51	0.26	9.14	1.39	11.23	1.77
	Y12	5.11	0.59	0.47	0.13	10.94	1.04	10.69	1.25
	Y13	5.07	1.06	0.28	0.13	10.94	1.56	10.16	1.03
	Y14	3.75	0.58	0.04	0.04	10.73	1.45	13.22	1.66
	Y15	3.32	0.36	3.14	0.72	7.64	1.10	11.05	1.15
	Y16	2.94	0.29	0.51	0.30	10.62	1.61	9.39	1.22
	Y17	2.99	0.36	1.87	0.49	9.47	1.34	10.17	1.26
	Y18	3.63	0.44	0.73	0.09	8.01	0.79	12.47	1.27
ICCVAM		6.07	1.00	3.87	0.53	8.49	1.35	7.51	1.09

property were selected seven times out of ten replications for CHR: Mouse: Liver Hypertrophy (Y1). (2D-topological, 2D-connectivity index) pair was selected seven times out of ten replications for CHR: Mouse: Tumorigen (Y4). 0D-constitutional, 2D-walk and path, 1D-functional group, and 2D-molecular property were selected seven times out of ten replications for DEV: Rabbit: General Fetal Weight Reduction (Y8). For DEV: Rabbit: Pregnancy Related Maternal Preg Loss (Y10), (0D-constitutional, 2D-connectivity index) pair was selected seven times. (0D-constitutional, 2D-functional group) pair was selected seven times for DEV: Rabbit Skeletal Axial (Y11). For DEV: Rat: Pregnancy Related Embryo Fetal Loss (Y13), 2D-connectivity index, (0D-constitutional, 2D-atom centered fragment) and (0D-constitutional, 2D-molecular property) were selected seven times out of ten replications.

By using the IDC method with random forests, 2D-topological and (2D-functional, 2D-atom centered fragment) were selected five times for DEV: Rat: Skeletal Axial (Y15), and 1D-functional group was selected five times for MGR: Rat: Viability PND4 (Y18). By using the IDC method with tree AdaBoost, 2-D topological was selected five times out of ten replications for CHR: Rat: Tumorigen (Y7). 2D-molecular property was selected six times for DEV: Rabbit: Pregnancy Related Maternal Preg Loss (Y10), and 2D-autocorrelation was selected five times for DEV: Rat: Skeletal Axial (Y15). In the ToxRefDB data set, we observed that the IDC methods with random forests and tree-based AdaBoost tend to select univariate feature categories more frequently than bivariate feature categories while the IDC method with SVM selected both univariate and bivariate feature categories.

2. ICCVAM data set. As shown in Tables 5.8, 5.9, and 5.10, the IDC methods with three classification inducers tend to select univariate feature categories more

frequently compared to bivariate feature categories. The 2D-molecular property was selected most frequently by all classification inducers (8 times out of ten replications with SVM; 9 times with random forests and with tree-based AdaBoost). 0D-constitutional (8 times with SVM; 6 times with random forests and AdaBoost), 2D-fingerprints (8 times with SVM; 6 times with AdaBoost), 1D-functional group (9 times with random forests; seven times with AdaBoost), and 2D-atom centered fragment (9 times with random forests) were also frequently selected in ICCVAM data analysis.

In summary, we observed that the IDC methods and IDC stacking methods can improve prediction accuracy by considering interactive effects among categories of feature variables in both data sets. It is interesting to note that both IDC and IDC stacking methods failed to improve classification performance for a few endpoints. As Shipp and Kuncheva [2002] noted, the decision committee method can perform worse than a single classifier due to dependency among base classifiers. Wang et al. [2009] also argued that the performance of the decision committee method depends on the data characteristics and showed through empirical experiments that the decision committee methods are not always better than a single classifier applying to SVM. Due to the complicated aggregation mechanism of the decision committee methods, it is not obvious why the IDC methods or the IDC stacking methods performed worse than single classifiers for a few endpoints. However, simulation studies in the previous section already showed that the IDC method can fail to improve classification performance in some cases. Also, it is not surprising that selecting base classifiers through forward selection with 5-fold CV worked better than stacked generalization in many endpoints as shown in this data example. It is possible that we can improve the classification performance of the IDC stacking method by finding a more sophisticated, optimized learning algorithm to learn an aggregation rule, as suggested by

Table 5.8: The total number of times each descriptor category or pair of descriptor categories (base classifiers) were selected in the final classifier over ten replications by the IDC method with SVM. The results of the best kernel and aggregation rule are presented.

Category	ToxRefDB																				ICCVAM
	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	Y9	Y10	Y11	Y12	Y13	Y14	Y15	Y16	Y17	Y18			
0D-constitutional	3	2	1	3	3	4	1	7	2	5	1	3	3	3	5	1	2	1			8
2D-topological	4	1	1	6	1	2	0	3	1	3	3	2	4	3	3	1	3	2			7
2D-walk and path	4	1	0	6	0	0	1	7	0	2	1	1	7	2	3	0	0	1			7
2D-connectivity Index	2	1	0	5	1	3	2	4	2	5	1	1	6	2	3	0	3	1			7
2D-autocorrelation	3	4	1	4	2	1	1	3	1	2	1	4	1	4	4	0	2	1			3
2D-eigenvalue based	3	4	0	3	1	0	1	5	1	2	2	1	3	1	2	0	1	3			5
1D-functional group	7	2	2	2	2	1	1	7	1	4	5	2	2	4	1	1	4	2			4
2D-atom centered fragment	4	1	0	0	2	0	3	2	0	2	3	1	1	2	1	2	0	1			5
2D-molecular property	7	2	0	4	2	1	1	7	0	6	1	1	4	1	1	2	2	3			8
2D-fingerprints	0	1	0	5	2	0	1	2	0	2	2	3	3	3	4	1	1	4			8
(0D-constitutional, 2D-topological)	4	2	1	6	0	0	0	3	1	4	1	1	5	0	3	0	1	1			3
(0D-constitutional, 2D-walk and path)	5	2	0	5	1	1	0	6	0	4	1	1	6	0	3	1	4	0			5
(0D-constitutional, 2D-connectivity Index)	2	2	2	5	0	1	0	4	1	7	0	2	3	0	1	0	2	2			1
(0D-constitutional, 2D-autocorrelation)	2	2	0	1	0	0	1	2	0	2	2	1	3	0	1	1	2	2			3
(0D-constitutional, 2D-eigenvalue based)	2	2	0	2	1	0	0	1	0	2	0	2	5	0	1	0	0	0			2
(0D-constitutional, 1D-functional group)	3	1	2	3	4	0	0	6	0	1	7	3	3	0	3	0	2	0			3
(0D-constitutional, 2D-atom centered fragment)	2	3	1	4	1	0	0	6	0	3	3	3	7	1	5	0	1	3			2
(0D-constitutional, 2D-molecular property)	6	2	0	2	1	0	0	2	0	4	0	0	7	0	1	0	3	0			6
(0D-constitutional, 2D-fingerprints)	2	5	0	4	2	1	1	1	0	2	4	3	3	1	4	0	2	4			2
(2D-topological, 2D-walk and path)	5	3	0	4	1	0	0	5	0	3	0	2	4	1	1	0	2	1			3
(2D-topological, 2D-connectivity Index)	3	1	0	7	2	0	1	4	0	2	0	2	3	1	0	0	0	0			1
(2D-topological, 2D-autocorrelation)	5	4	0	3	1	0	0	3	0	3	0	1	6	1	3	0	0	1			3
(2D-topological, 2D-eigenvalue based)	2	1	0	3	1	0	0	1	0	0	1	1	4	0	4	0	0	1			2
(2D-topological, 1D-functional group)	3	2	2	4	2	0	2	4	1	1	2	0	5	1	2	0	0	0			3
(2D-topological, 2D-atom centered fragment)	1	5	0	3	2	0	0	5	1	4	1	5	2	1	1	0	1	1			1
(2D-topological, 2D-molecular property)	3	0	0	1	0	0	0	3	0	3	1	3	5	1	1	0	1	1			2
(2D-topological, 2D-fingerprints)	1	3	2	2	3	0	2	3	1	2	1	4	4	3	4	1	4	4			2
(2D-walk and path, 2D-connectivity Index)	0	0	0	5	2	0	0	4	0	5	2	1	5	0	0	0	2	0			3
(2D-walk and path, 2D-autocorrelation)	3	2	0	2	1	0	0	1	0	5	2	2	3	1	1	0	0	0			0
(2D-walk and path, 2D-eigenvalue based)	2	4	0	4	1	0	0	3	0	1	0	0	4	1	0	0	2	1			1
(2D-walk and path, 1D-functional group)	4	4	1	3	2	0	0	5	0	1	2	2	2	1	5	0	2	0			6
(2D-walk and path, 2D-atom centered fragment)	3	1	1	3	1	0	2	2	0	2	1	3	6	2	1	1	4	1			4
(2D-walk and path, 2D-molecular property)	3	1	0	2	2	0	0	3	0	3	1	1	4	1	0	1	3	1			2
(2D-walk and path, 2D-fingerprints)	1	2	1	3	0	0	3	2	0	2	2	3	5	0	2	1	1	2			5
(2D-connectivity Index, 2D-autocorrelation)	5	1	0	3	0	0	0	2	0	3	1	0	2	0	2	0	3	2			1
(2D-connectivity Index, 2D-eigenvalue based)	2	2	0	1	1	0	0	2	1	0	1	1	3	3	1	0	0	2			1
(2D-connectivity Index, 1D-functional group)	6	2	0	5	2	0	0	3	0	2	3	1	3	1	3	0	2	0			4
(2D-connectivity Index, 2D-atom centered fragment)	1	3	1	1	2	0	0	3	1	2	4	3	5	3	2	1	2	2			4
(2D-connectivity Index, 2D-molecular property)	1	1	0	4	0	0	0	1	0	2	0	2	4	2	1	0	1	2			2
(2D-connectivity Index, 2D-fingerprints)	1	0	0	5	0	0	0	3	1	4	5	5	3	1	3	0	2	0			3
(2D-autocorrelation, 2D-eigenvalue based)	2	2	0	3	3	0	0	3	0	0	0	2	3	2	0	0	2	2			2
(2D-autocorrelation, 1D-functional group)	3	2	0	3	0	0	2	2	0	1	1	0	2	2	2	0	4	3			0
(2D-autocorrelation, 2D-atom centered fragment)	1	0	0	3	2	0	0	3	0	1	3	1	2	3	2	0	0	3			0
(2D-autocorrelation, 2D-molecular property)	5	1	0	1	1	0	0	4	0	3	0	0	2	0	1	0	1	2			1
(2D-autocorrelation, 2D-fingerprints)	0	0	0	1	0	0	0	2	0	1	1	0	3	2	1	0	3	1			3
(2D-eigenvalue based, 1D-functional group)	1	0	0	2	1	0	0	2	0	1	1	1	4	0	0	0	2	2			1
(2D-eigenvalue based, 2D-atom centered fragment)	0	1	0	3	1	0	0	2	0	1	1	1	2	1	2	0	1	0			2
(2D-eigenvalue based, 2D-molecular property)	4	0	0	3	0	0	0	1	0	0	1	1	4	1	1	0	2	4			2
(2D-eigenvalue based, 2D-fingerprints)	1	1	1	5	1	0	0	2	0	1	1	0	2	2	2	0	2	1			2
(1D-functional group, 2D-atom centered fragment)	2	1	0	2	0	1	1	2	0	1	2	2	2	2	3	1	1	1			3
(1D-functional group, 2D-molecular property)	4	2	0	2	1	0	0	2	1	1	3	2	1	5	1	0	3	3			5
(1D-functional group, 2D-fingerprints)	0	0	0	1	0	1	0	4	0	1	1	2	2	1	3	0	4	0			3
(2D-atom centered fragment, 2D-molecular property)	1	4	0	5	0	0	0	3	1	2	3	1	6	4	0	0	3	3			2
(2D-atom centered fragment, 2D-fingerprints)	0	1	0	1	0	0	1	4	0	0	2	2	3	2	0	0	1	2			3
(2D-molecular property, 2D-fingerprints)	1	0	0	4	0	0	2	0	0	0	1	0	5	1	1	0	2	2			3

Table 5.9: The total number of times each descriptor category or pair of descriptor categories (base classifiers) were selected in the final classifier over ten replications by the IDC method with random forests.

Category	ToxRefDB																			ICCVAM
	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	Y9	Y10	Y11	Y12	Y13	Y14	Y15	Y16	Y17	Y18		
0D-constitutional	0	2	2	2	0	0	2	0	0	0	0	0	0	1	1	1	3	0		6
2D-topological	1	1	4	0	2	0	1	0	1	4	1	0	1	1	5	1	0	0		3
2D-walk and path	1	2	1	2	0	3	2	2	1	2	1	0	1	0	1	1	1	1		2
2D-connectivity Index	2	3	3	3	1	0	3	1	0	2	3	2	1	1	1	0	3	0		4
2D-autocorrelation	0	1	2	3	0	1	3	0	0	3	0	3	1	0	3	1	1	3		5
2D-eigenvalue based	1	3	3	1	0	1	4	1	0	3	1	0	3	3	0	2	1	2		5
1D-functional group	0	3	0	1	0	3	3	2	2	2	2	1	2	1	1	2	1	5		9
2D-atom centered fragment	1	0	0	1	2	0	1	2	1	2	1	4	1	0	4	0	1	0		9
2D-molecular property	1	3	1	3	1	1	3	2	1	4	3	0	2	2	2	0	0	4		9
2D-fingerprints	1	1	1	2	0	2	1	0	0	2	2	3	0	0	1	0	1	0		4
(0D-constitutional, 2D-topological)	0	0	0	0	0	0	0	1	1	0	1	2	3	0	2	0	1	0		0
(0D-constitutional, 2D-walk and path)	0	1	0	0	0	1	0	0	1	1	0	0	1	1	2	0	0	1		0
(0D-constitutional, 2D-connectivity Index)	3	2	1	0	1	0	0	0	0	1	1	0	0	0	1	0	0	0		0
(0D-constitutional, 2D-autocorrelation)	0	1	0	0	0	1	1	1	0	2	0	1	2	1	0	0	2	1		0
(0D-constitutional, 2D-eigenvalue based)	0	1	0	0	1	0	0	1	2	1	1	0	0	1	0	1	2	0		0
(0D-constitutional, 1D-functional group)	2	1	1	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0		2
(0D-constitutional, 2D-atom centered fragment)	1	0	0	0	0	0	1	0	0	1	0	0	0	0	2	1	0	1		0
(0D-constitutional, 2D-molecular property)	0	1	2	0	0	2	0	0	0	1	1	0	0	0	0	1	0	1		1
(0D-constitutional, 2D-fingerprints)	0	0	2	1	1	0	0	0	1	0	0	1	0	0	1	0	2	0		1
(2D-topological, 2D-walk and path)	1	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0		0
(2D-topological, 2D-connectivity Index)	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0		1
(2D-topological, 2D-autocorrelation)	1	1	0	0	0	1	1	0	0	0	1	0	0	0	2	0	1	0		2
(2D-topological, 2D-eigenvalue based)	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0		1
(2D-topological, 1D-functional group)	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1		0
(2D-topological, 2D-atom centered fragment)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	1		0
(2D-topological, 2D-molecular property)	3	1	0	0	1	1	2	0	2	2	0	0	0	1	0	0	0	0		0
(2D-topological, 2D-fingerprints)	1	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0		1
(2D-walk and path, 2D-connectivity Index)	0	2	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	1		1
(2D-walk and path, 2D-autocorrelation)	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	1	0		1
(2D-walk and path, 2D-eigenvalue based)	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	1	0		0
(2D-walk and path, 1D-functional group)	0	1	0	0	1	0	1	0	2	1	1	1	0	0	0	1	0	1		0
(2D-walk and path, 2D-atom centered fragment)	1	0	0	2	1	0	2	0	0	0	0	0	0	0	3	1	0	0		0
(2D-walk and path, 2D-molecular property)	0	1	0	1	1	1	2	0	0	2	1	1	0	0	1	0	0	1		1
(2D-walk and path, 2D-fingerprints)	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	1	1	0		0
(2D-connectivity Index, 2D-autocorrelation)	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	1	0		1
(2D-connectivity Index, 2D-eigenvalue based)	0	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0		0
(2D-connectivity Index, 1D-functional group)	0	1	0	0	1	0	0	1	0	1	0	1	0	0	1	0	2	0		1
(2D-connectivity Index, 2D-atom centered fragment)	0	0	0	0	0	0	2	0	0	0	0	1	0	0	0	0	0	0		1
(2D-connectivity Index, 2D-molecular property)	0	2	0	1	1	0	0	0	1	1	0	0	0	0	1	0	1	0		0
(2D-connectivity Index, 2D-fingerprints)	0	1	1	0	0	1	0	0	0	2	0	0	0	0	1	0	1	0		0
(2D-autocorrelation, 2D-eigenvalue based)	0	0	0	1	1	0	0	0	0	1	0	0	0	1	0	1	1	0		0
(2D-autocorrelation, 1D-functional group)	1	0	0	1	0	1	1	0	0	1	0	1	0	1	2	0	0	0		2
(2D-autocorrelation, 2D-atom centered fragment)	0	1	0	0	1	1	0	0	0	2	0	0	0	0	0	0	0	0		0
(2D-autocorrelation, 2D-molecular property)	0	0	0	0	0	0	0	0	0	2	0	0	0	0	3	0	1	1		1
(2D-autocorrelation, 2D-fingerprints)	0	0	1	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0		1
(2D-eigenvalue based, 1D-functional group)	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	2	0		0
(2D-eigenvalue based, 2D-atom centered fragment)	0	2	0	1	0	0	0	0	0	0	0	0	1	0	1	0	1	0		0
(2D-eigenvalue based, 2D-molecular property)	0	1	1	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0		0
(2D-eigenvalue based, 2D-fingerprints)	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0		0
(1D-functional group, 2D-atom centered fragment)	2	1	0	3	1	0	2	1	0	1	0	1	0	1	5	2	1	0		1
(1D-functional group, 2D-molecular property)	0	2	0	1	1	0	1	0	2	0	1	0	0	2	0	0	2	2		4
(1D-functional group, 2D-fingerprints)	0	0	0	1	0	1	2	1	1	0	0	0	0	0	1	0	0	0		0
(2D-atom centered fragment, 2D-molecular property)	0	1	0	0	0	0	3	1	0	1	0	0	1	0	0	1	0	0		1
(2D-atom centered fragment, 2D-fingerprints)	0	0	1	2	0	0	1	0	0	0	0	0	0	0	0	0	0	0		0
(2D-molecular property, 2D-fingerprints)	1	0	1	0	0	0	1	0	1	1	0	2	0	0	0	0	0	1		2

Table 5.10: The total number of times each descriptor category or pair of descriptor categories (base classifiers) were selected in the final classifier over ten replications by the IDC method with tree AdaBoost.

Category	ToxRefDB																			ICCVAM
	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	Y9	Y10	Y11	Y12	Y13	Y14	Y15	Y16	Y17	Y18		
0D-constitutional	0	2	0	1	1	0	2	0	1	1	0	1	0	0	1	1	1	1		6
2D-topological	1	2	3	3	0	0	5	0	0	3	0	3	1	1	2	1	4	2		5
2D-walk and path	0	3	2	0	0	2	3	2	1	5	3	0	2	0	3	0	2	0		2
2D-connectivity Index	1	2	2	2	0	0	2	0	0	2	1	2	1	2	3	0	4	2		7
2D-autocorrelation	1	1	3	2	1	2	3	0	0	2	0	2	0	1	5	0	1	4		4
2D-eigenvalue based	0	0	1	3	1	1	1	1	1	2	0	1	1	2	2	1	0	2		4
1D-functional group	1	1	0	0	1	2	1	1	0	2	2	0	0	0	1	2	2	1		7
2D-atom centered fragment	2	2	0	0	1	0	0	1	1	0	0	3	0	0	0	1	4	0		8
2D-molecular property	0	4	1	3	1	0	3	0	2	6	2	1	1	2	3	1	1	1		9
2D-fingerprints	0	0	1	0	0	2	4	0	0	3	1	2	0	0	1	1	0	0		6
(0D-constitutional, 2D-topological)	0	1	1	0	0	0	1	0	1	0	0	2	2	1	2	1	0	0		0
(0D-constitutional, 2D-walk and path)	1	1	0	2	1	0	1	1	1	1	0	1	1	0	1	0	0	0		0
(0D-constitutional, 2D-connectivity Index)	2	1	1	1	2	1	0	1	0	1	1	0	1	0	0	0	2	0		0
(0D-constitutional, 2D-autocorrelation)	3	0	1	0	1	1	2	1	0	4	1	1	0	0	1	0	2	2		2
(0D-constitutional, 2D-eigenvalue based)	1	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1		0
(0D-constitutional, 1D-functional group)	1	1	0	0	1	0	0	1	0	4	0	0	0	0	3	0	0	0		0
(0D-constitutional, 2D-atom centered fragment)	0	0	1	0	0	0	1	0	0	1	0	1	0	0	2	0	0	1		0
(0D-constitutional, 2D-molecular property)	0	0	0	3	0	0	1	0	1	1	2	0	0	1	1	0	0	0		0
(0D-constitutional, 2D-fingerprints)	1	0	0	1	1	0	0	1	0	2	0	0	1	0	3	1	3	0		1
(2D-topological, 2D-walk and path)	0	0	0	0	1	0	1	0	2	1	2	0	2	2	0	1	2	0		0
(2D-topological, 2D-connectivity Index)	1	0	0	2	0	0	0	0	0	2	0	0	0	0	0	0	1	2		1
(2D-topological, 2D-autocorrelation)	0	0	0	0	1	2	1	0	1	1	0	2	0	0	0	0	2	2		0
(2D-topological, 2D-eigenvalue based)	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	2	0		0
(2D-topological, 1D-functional group)	2	3	0	2	0	0	0	0	0	0	0	1	1	0	1	1	0	0		2
(2D-topological, 2D-atom centered fragment)	0	0	0	1	1	0	0	0	0	0	0	1	0	0	1	0	1	0		2
(2D-topological, 2D-molecular property)	0	1	1	1	0	2	1	0	0	0	0	0	0	0	0	0	0	0		1
(2D-topological, 2D-fingerprints)	1	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	2	1		1
(2D-walk and path, 2D-connectivity Index)	0	1	0	0	0	0	1	0	0	3	0	1	1	0	0	0	1	1		0
(2D-walk and path, 2D-autocorrelation)	0	1	0	1	1	0	0	0	0	2	0	1	1	0	1	1	0	0		0
(2D-walk and path, 2D-eigenvalue based)	1	0	0	2	0	0	0	0	0	2	0	0	0	0	1	0	1	0		0
(2D-walk and path, 1D-functional group)	0	0	1	0	1	0	0	0	0	2	1	1	0	0	0	1	0	0		2
(2D-walk and path, 2D-atom centered fragment)	0	0	0	0	0	0	0	0	1	1	2	0	0	2	0	1	0	0		0
(2D-walk and path, 2D-molecular property)	0	2	0	2	0	0	1	0	1	1	0	0	0	0	1	0	1	0		2
(2D-walk and path, 2D-fingerprints)	0	1	1	0	0	0	0	0	0	5	0	0	0	0	2	0	1	0		0
(2D-connectivity Index, 2D-autocorrelation)	0	1	1	0	1	1	2	1	0	1	0	2	0	1	4	0	1	0		0
(2D-connectivity Index, 2D-eigenvalue based)	0	0	0	0	0	0	1	1	0	3	0	0	1	1	0	0	1	0		1
(2D-connectivity Index, 1D-functional group)	0	1	0	1	1	1	0	0	0	2	0	1	0	1	0	0	2	0		1
(2D-connectivity Index, 2D-atom centered fragment)	0	0	0	0	0	1	0	1	0	0	0	0	0	0	2	0	1	0		1
(2D-connectivity Index, 2D-molecular property)	0	0	0	1	0	0	0	0	1	2	0	2	1	0	0	0	1	0		0
(2D-connectivity Index, 2D-fingerprints)	1	1	1	1	0	2	1	1	1	1	1	1	0	0	1	0	2	0		0
(2D-autocorrelation, 2D-eigenvalue based)	0	0	0	1	0	1	0	0	0	1	0	0	0	0	1	0	3	0		0
(2D-autocorrelation, 1D-functional group)	0	1	2	0	1	0	1	0	0	0	0	0	1	0	2	1	2	1		0
(2D-autocorrelation, 2D-atom centered fragment)	0	0	0	0	0	1	1	0	1	1	0	2	0	0	2	1	0	0		0
(2D-autocorrelation, 2D-molecular property)	1	1	0	0	0	0	1	0	1	0	0	0	0	0	2	1	3	1		3
(2D-autocorrelation, 2D-fingerprints)	1	0	0	0	0	0	1	0	0	2	0	1	0	0	3	0	2	0		0
(2D-eigenvalue based, 1D-functional group)	0	1	0	0	2	0	0	1	0	1	0	0	0	0	0	0	1	0		2
(2D-eigenvalue based, 2D-atom centered fragment)	1	2	0	1	0	0	0	0	1	0	0	0	0	1	0	1	0	0		0
(2D-eigenvalue based, 2D-molecular property)	0	2	1	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0		0
(2D-eigenvalue based, 2D-fingerprints)	0	1	0	1	0	0	0	0	0	0	0	2	0	0	2	0	0	0		0
(1D-functional group, 2D-atom centered fragment)	1	0	0	1	1	0	0	0	0	1	0	1	1	0	2	0	0	2		1
(1D-functional group, 2D-molecular property)	0	1	0	2	2	0	2	0	1	1	0	1	0	0	1	0	0	1		3
(1D-functional group, 2D-fingerprints)	0	0	1	2	1	0	1	1	1	2	1	0	0	0	0	2	0	0		0
(2D-atom centered fragment, 2D-molecular property)	0	0	0	1	0	0	4	1	0	0	0	1	0	0	1	1	0	0		2
(2D-atom centered fragment, 2D-fingerprints)	1	1	0	3	0	1	3	2	0	0	0	2	0	0	2	1	2	0		0
(2D-molecular property, 2D-fingerprints)	0	2	0	0	0	1	3	0	1	1	0	0	0	0	0	0	1	0		2

Wolpert [1992].

Chapter 6

Discussion

6.1 The change-line method

In this section, we summarize our research on the change-line method, and we briefly discuss other issues related to the change-line classification and regression method.

6.1.1 Summary

In this dissertation, we have presented a new method of classification and regression by finding a line that can divide a whole observation into two heterogeneous subgroups. This method is based on the assumption that there exist two latent groups in a whole population, each of which yields a response having a different probability distribution or having a different association with a covariate of interest. The applicability of this method was illustrated with a simulation study and with an example of chemical toxicity data. In the simulation study, we demonstrated that the proposed method works well under the two latent subgroups assumption. A small investigation for the statistical hypothesis testing for the presence of a change-line was carried out. The programs for both the simulation study and the example of chemical toxicity are written in Microsoft Visual $C++$.

The attractiveness of the presented method is threefold. First, we can consider more flexible models by allowing the response variable to be either continuous or discrete. Second, the computational cost of separating a population into two subgroups is reduced by use of an efficient algorithm. Third, the proposed method combines the idea of machine learning with statistical modeling, so it gives us more reproducible results compared to the pure machine learning approach.

6.1.2 Generalization to the probabilistic model

Method

In this section, we consider a generalization of the change-line models to the probabilistic model. Suppose the true model is defined by

$$Y(X,Z;\theta) \sim C(X;\omega,\gamma)F_1(Z;\beta,\tau) + \{1 - C(X;\omega,\gamma)\}F_2(Z;\delta,\tau), \tag{6.1}$$

where $C(X;\omega,\gamma) = \mathbf{1}\{\omega^T X - \gamma > 0\}$. We consider the working model as follows:

$$Y(X,Z;\theta) \sim G(X;b,a)F_1(Z;\beta,\tau) + \{1 - G(X;a,b)\}F_2(Z;\delta,\tau), \tag{6.2}$$

where $G(X;a,b) = \frac{e^{a+b^T X}}{1+e^{a+b^T X}}$, $a + b^T X = \kappa(\omega^T X - \gamma)$, and $\kappa > 0$. As $k \to \infty$, note that $\frac{e^{a+b^T X_i}}{1+e^{a+b^T X_i}} \to 1$ for $\omega^T X_i - \gamma > 0$ while $\frac{e^{a+b^T X_i}}{1+e^{b^T X_i+a}} \to 0$ for $\omega^T X_i - \gamma \leq 0$. We replace the indicator function $C(X;\omega,\gamma) = \mathbf{1}\{\omega^T X - \gamma > 0\}$ with a type of logistic function $G(X;a,b) = \frac{e^{a+b^T X}}{1+e^{a+b^T X}}$ that is smoother than the indicator function. Note that we only observe $Y^* = (Y,X,Z)$, so we can consider C to be a missing class for each individual. Let W denote complete data (Y,X,Z,C). In this frame, the new parameter of interest

$\theta = (\theta_1, \theta_2)$, where $\theta_1 = (a, b_1, b_2)$, and $\theta_2 = (\mu_0, \mu_1, \sigma_0^2, \sigma_1^2)$. Then completed log likelihood can be written as follows:

$$l(\theta|W) = \sum_{i=1}^{n} [c_i \log g_{\theta_1}(x_i) f_1(y_i; \theta_2) + (1 - c_i) \log (1 - g_{\theta_1}(x_i)) f_0(y_i : \theta_2)], \qquad (6.3)$$

where

$$\begin{aligned} g_{\theta_1}(x) &= \frac{e^{a+b^T x}}{1 + e^{a+b^T x}}, \\ f_0(y; \theta_2) &= \frac{1}{\sqrt{2\pi\sigma_0^2}} \exp\left\{\frac{(y-\mu_0)^2}{-2\sigma_0^2}\right\}, \\ f_1(y; \theta_2) &= \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp\left\{\frac{(y-\mu_1)^2}{-2\sigma_1^2}\right\}. \end{aligned}$$

Having an individual class of c_i as missing data, we can utilize the Expectation-Maximization (EM) algorithm to estimate the parameter θ. Note that score equations are given as follows:

$$\begin{aligned} \frac{\partial \ell_\theta(w)}{\partial a} &= \sum_{i=1}^{n} (c_i - g_i) = 0 \\ \frac{\partial \ell_\theta(w)}{\partial b_j} &= \sum_{i=1}^{n} (c_i - g_i) x_{ij} = 0, j = 1, 2 \\ \frac{\partial \ell_\theta(w)}{\partial \mu_0} &= \sum_{i=1}^{n} (1 - c_i)(y_i - \mu_0) = 0 \\ \frac{\partial \ell_\theta(w)}{\partial \mu_1} &= \sum_{i=1}^{n} c_i (y_i - \mu_1) = 0 \\ \frac{\partial \ell_\theta(w)}{\partial \sigma_0^2} &= \sum_{i=1}^{n} (1 - c_i)((y_i - \mu_0)^2 - \sigma_0^2) = 0 \\ \frac{\partial \ell_\theta(w)}{\partial \sigma_1^2} &= \sum_{i=1}^{n} c_i ((y_i - \mu_1)^2 - \sigma_1^2) = 0. \end{aligned}$$

1. E-step

In the E-step of the k^{th} iteration, the conditional expectation can be calculated by

$$E(c_i|y_i, x_i, \theta^{(k)}) = \frac{g_i(\theta_1^{(k)}) f_{1i}(\theta_{2,1}^{(k)})}{g_i(\theta_1^{(k)}) f_{1i}(\theta_{2,1}^{(k)}) + (1 - g_i(\theta_1^{(k)})) f_{0i}(\theta_{2,0}^{(k)})},$$

where

$$\begin{aligned} g_i(\theta_1^{(k)}) &= \frac{e^{a^{(k)} + b^{(k)T} x}}{1 + e^{a^{(k)} + b^{(k)T} x}}, \\ f_{1i}(\theta_{2,1}^{(k)}) &= \frac{1}{\sqrt{2\pi\sigma_1^{2,(k)}}} \exp\left\{ \frac{(y - \mu_1^{(k)})^2}{-2\sigma_1^{2,(k)}} \right\}, \\ f_{0i}(\theta_{2,0}^{(k)}) &= \frac{1}{\sqrt{2\pi\sigma_0^{2,(k)}}} \exp\left\{ \frac{(y - \mu_0^{(k)})^2}{-2\sigma_0^{2,(k)}} \right\}. \end{aligned}$$

2. M-step

In the M-step of the k^{th} iteration, we solve the conditional equations to obtain k^{th} estimators by

$$\begin{aligned} a^{(k+1)} &= a^{(k)} + [\sum_{i=1}^{n} g_i(a^{(k)}, b^{(k)})(1 - g_i(a^{(k)}, b^{(k)}))]^{-1} \\ &\times [\sum_{i=1}^{n} (E(c_i|\cdot) - g_i(a^{(k)}, b^{(k)}))] \\ b_j^{(k+1)} &= b_j^{(k)} + [\sum_{i=1}^{n} g_i(a^{(k)}, b^{(k)})(1 - g_i(a^{(k)}, b^{(k)})) x_{ij}^2]^{-1} \\ &\times [\sum_{i=1}^{n} (E(c_i|\cdot) - g_i(a^{(k)}, b^{(k)})) x_{ii}], \ j = 1, 2. \end{aligned}$$

$$
\begin{aligned}
\mu_0^{(k+1)} &= \mu_0^{(k)} + \frac{\sum_{i=1}^{n}(1 - E(c_i|\cdot)y_i}{\sum_{i=1}^{n}(1 - E(c_i|\cdot)))} \\
\mu_1^{(k+1)} &= \mu_1^{(k)} + \frac{\sum_{i=1}^{n} E(c_i|.)y_i}{\sum_{i=1}^{n} E(c_i|\cdot)} \\
\sigma_0^{2,(k+1)} &= \sigma_0^{2,(k)} + \frac{\sum_{i=1}^{n}(1 - E(c_i|\cdot))(y_i - \mu_0^{(k)})^2}{\sum_{i=1}^{n}(1 - E(c_i|\cdot))} \\
\sigma_1^{2,(k+1)} &= \sigma_1^{2,(k)} + \frac{\sum_{i=1}^{n} E(c_i|\cdot)(y_i - \mu_1^{(k)})^2}{\sum_{i=1}^{n} E(c_i|\cdot)}.
\end{aligned}
$$

The requirement that $\|\omega\| = 1$ enables us to calculate κ by $\|\hat{b}\| = \sqrt{\hat{b}_1^2 + \hat{b}_2^2}$. We repeat the E and M steps until the difference between estimates in $(k-1)^{th}$ iteration and k^{th} iteration is smaller than a pre-determined threshold value.

Preliminary simulation study

We conducted simulation studies to see if the proposed working model works well in the change-line classification problem. The true value for parameters in the true models are given as $\omega_0 = (-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$, $\gamma_0 = 0$, $(\mu_0, \mu_1) = (2, 0)$, and $(\sigma_0^2, \sigma_1^2) = (4, 1)$. The sample sizes we explored were (200, 600, and 6000). We repeated the EM algorithm iterations until the $(k-1)^{th}$ estimates and k^{th} estimates were less than or equal to the given convergent criterion (0.0001 for the large sample and 0.5 for the small sample) or until the maximum number of iterations (100). As shown in Table 6.1, the estimated parameters for both change-line parameters and model parameters are very close to the true values with very small MC variations. The main advantages of applying the probabilistic working model with the EM algorithm are that we can run the change-line method on large data sets (larger sample size of 6000), and computation is significantly faster than with the deterministic change-line models.

Table 6.1: Summary statistics from 100 replications of simulation study for the change-line classification using probabilistic working model and the EM algorithm. For all simulations, true values of the parameters were chosen as $(\omega_1^0 = -1/\sqrt{2}, \omega_2^0 = 1/\sqrt{2}, \gamma^0 = 0, \mu_1^0 = 0, \mu_0^0 = 2, \sigma_1^{2,0} = 1, \sigma_0^{2,0} = 4)$.

Parameter		Sample Size		
		200	600	6000
$\omega_1 = \mathbf{-0.707}$	MCRM	-0.706	-0.708	-0.707
	MCSE	0.004	0.001	0.000
$\omega_2 = \mathbf{0.707}$	MCRM	0.709	0.706	0.707
	MCSE	0.004	0.001	0.000
$\gamma = \mathbf{0}$	MC Mean	0.033	0.014	0.010
	MCSE	0.005	0.002	0.001
k	MC Mean	11.239	28.218	25.267
	MCSE	0.599	1.089	0.254
$\mu_0 = \mathbf{2}$	MC Mean	2.064	2.048	2.016
	MCSE	0.018	0.011	0.004
$\mu_1 = \mathbf{0}$	MC Mean	-0.028	-0.011	-0.005
	MCSE	0.010	0.005	0.002
$\sigma_0^2 = \mathbf{4}$	MC Mean	3.821	3.951	4.006
	MCSE	0.063	0.032	0.010
$\sigma_1^2 = \mathbf{1}$	MC Mean	0.953	0.987	1.001
	MCSE	0.014	0.008	0.003

Note that the EM algorithm returns local maximizers, thus EM might not converge to the true model with bad initial values ([Karlis and Xekalaki, 2003]; [Nader et al., 2011]). Therefore, we investigated the influence of choosing initial values on estimations through simulation studies in the probabilistic working model to solve the change-line classification problem. For all simulations, subgroup 0 satisfying $(\omega^T X - \gamma > 0)$ has a smaller mean (0) and variance (1) than subgroup 1 satisfying $(\omega^T X - \gamma \leq 0)$ (mean 2, variance 4). In Simulations 1 and 2, initial values for model parameters in subgroup 0 were smaller than those in subgroup 1. Different initial values for change-line parameters were given to Simulations 1 and 2. In Simulation 3, the same sets of initial values were given for two subgroups. In Simulation 4, greater initial values for mean and variance were given to subgroup 0, which is opposite to the true model.

Simulation 1: $b^0 = (\cos(u), \sin(u))$, where $u \sim unif(0, \pi)$, $a^0 = 0$, $\mu^0 = (0, 0.1)$, and $\sigma^{2,0} = (1, 2)$.

Simulation 2: $b^0 = (\cos(u), \sin(u))$, where $u \sim unif(100, 300)$, $a^0 = 0$, $\mu^0 = (0, 0.1)$, and $\sigma^{2,0} = (1, 2)$.

Simulation 3: $b^0 = (\cos(u), \sin(u))$, where $u \sim unif(100, 300)$, $a^0 = 0$, $\mu^0 = (0, 0)$, and $\sigma^{2,0} = (1, 1)$.

Simulation 4: $b^0 = (\cos(u), \sin(u))$, where $u \sim unif(100, 300)$, $a^0 = 0$, $\mu^0 = (0.1, 0)$, and $\sigma^{2,0} = (2, 1)$.

The simulation results are given in Table 6.2. The estimates in Simulations 1 and 2 were quite close to the true values of parameters. The estimates in Simulation 3 were worse than those in Simulations 1 and 2 although the estimated mean and variance of subgroup 0 were smaller than those of subgroup 1. In Simulation 4, the mean and variance of subgroup 0 were still smaller than those of subgroup 1, but quite far from

Table 6.2: Summary statistics from 100 replications of simulation study for the change-line classification using probabilistic working model and the EM algorithm under different initial values. For all simulations, true values of the parameters were chosen as $(\omega_1^0 = -1/\sqrt{2}, \omega_2^0 = 1/\sqrt{2}, \gamma^0 = 0, \mu_0^0 = 0, \mu_1^0 = 2, \sigma_0^{2,0} = 1, \sigma_1^{2,0} = 4)$. Sample size was 300.

Parameter		Initial value			
		Sim1	Sim2	Sim3	Sim4
$\omega_1 = \mathbf{-0.707}$	MCRM	-0.708	-0.708	-0.161	0.597
	MCSE	0.001	0.001	0.073	0.070
$\omega_2 = \mathbf{0.707}$	MCRM	0.706	0.706	0.987	0.802
	MCSE	0.001	0.001	0.038	0.042
$\gamma = \mathbf{0}$	MC Mean	0.015	0.015	-0.004	-0.040
	MCSE	0.003	0.003	0.005	0.008
k	MC Mean	31.335	31.363	13.800	11.862
	MCSE	2.359	2.357	0.617	1.018
$\mu_0 = \mathbf{0}$	MC Mean	-0.010	-0.010	0.407	0.874
	MCSE	0.006	0.006	0.059	0.055
$\mu_1 = \mathbf{2}$	MC Mean	2.042	2.042	1.622	1.150
	MCSE	0.012	0.012	0.061	0.056
$\sigma_0^2 = \mathbf{1}$	MC Mean	0.984	0.984	1.971	3.060
	MCSE	0.009	0.009	0.137	0.125
$\sigma_1^2 = \mathbf{4}$	MC Mean	3.928	3.928	3.638	3.381
	MCSE	0.035	0.035	0.057	0.055

the true values. This experiment suggests that estimates by an EM algorithm for the probabilistic working model do not heavily depend on choosing initial values for the change-line parameters but might depend on choosing initial values for the model parameters. Therefore, care should be taken to choose initial values in order to utilize the EM algorithm for practical application of the probabilistic working model in the change-line problem.

Example: Chemical toxicity data

We applied the generalized method to a chemical toxicity data set to find two subgroups of chemical compounds with different means and variances of toxicity activity. As Karlis and Xekalaki [2003] stated, it is a natural choice to start with estimates obtained by another method. Therefore, we applied the original change-line classification model for 591 randomly selected chemical compounds among 5,917 (about 10%), and we started with the estimates obtained. Estimated initial values for model parameters were $(\mu_0^0, \mu_1^0) = (2.30734, 3.21580)$, and $(\sigma_0^{2,0}, \sigma_1^{2,0}) = (0.50692, 0.93877)$. To compare the results with the original change-line classification, we applied the generalized method for 1,000 and 3,000 subsamples instead of all remaining compounds although we could apply the generalized model to a large data set. The stopping criterion was given as either a convergent criterion of $0.01 \sim 0.0001$ or when a maximum number of iterations $300 \sim 3,000$ are satisfied. (μ_0, σ_0^2) denotes mean and variance for a subgroup of chemicals satisfying $\mathbf{1}\{\omega^T X - \gamma \leq 0\}$, and (μ_1, σ_1^2) denotes mean and variance for a subgroup of chemicals satisfying $\mathbf{1}\{\omega^T X - \gamma > 0\}$.

Table 6.3 shows the results of applying the generalized method with the EM algorithm. LSE denotes the estimates by using the original (deterministic) change-line classification model. We observed that estimates by the probabilistic working model with the EM algorithm were quite close to the estimates obtained by the deterministic change-line model with smaller MCSE. Considering the computational cost, generalization of the change-line method to the probabilistic model using the EM algorithm appears to work well if we start with good initial values for regression parameters.

Table 6.3: Summary statistics from 100 replications of applying the probabilistic working model with the EM algorithm to solve change-line classification problem in chemical toxicity data. LSE denotes results obtained by the deterministic change-line model, and EM denotes results obtained by the probabilistic working model with the EM algorithm.

		LSE	EM	
Parameter		1000	1000	3000
ω_1	MCRM	0.064	0.0725	0.0689
	MCSE	0.003	0.0021	0.0009
ω_2	MCRM	0.998	0.9972	0.9976
	MCSE	0.000	0.0002	0.0001
γ	MC Mean	0.000	-0.0350	-0.0320
	MCSE	0.004	0.0018	0.0008
κ	MC Mean		13.0745	12.7965
	MCSE		0.2288	0.0875
μ_0	MC Mean	2.220	2.1008	2.1006
	MCSE	0.008	0.0032	0.0016
μ_1	MC Mean	3.078	3.3087	3.2983
	MCSE	0.014	0.0072	0.0029
σ_0^2	MC Mean	0.495	0.3143	0.3133
	MCSE	0.006	0.0028	0.0011
σ_1^2	MC Mean	1.052	0.9502	0.9498
	MCSE	0.011	0.0067	0.0029

6.2 Weak convergence of the change-line regression

In Chapter 4, the consistency and the rates of convergence of M-estimators in the change-line regression model were studied through empirical process techniques. In this section, we briefly discuss the weak convergence of M-estimators for both model parameters $\hat{\varphi} = (\hat{\beta}, \hat{\delta})$ and change-line parameters $\hat{\zeta} = (\hat{\omega}, \hat{\gamma})$. Note that $\|\omega\| = 1$, where $(\omega_1, \omega_2) = (\cos(\alpha), \sin(\alpha))$. Therefore, we study the limiting distribution of the angle $\hat{\alpha}$ instead of the direction vector $\hat{\omega}$. Kosorok and Song [2007] proved that $\sqrt{n}(\hat{\varphi} - \varphi_0)$, which are regular parameter parts in the one-dimensional change-point regression model, is asymptotically linear, converging weakly in the uniform norm to a tight, mean zero Gaussian process. For the weak convergence of the change-point parameter, they proved that the re-parameterized process $Q_n(h) = M_n(\gamma_0 + \frac{h}{n})$ converges weakly to a right-continuous jump process $Q(h)$ in some Skorohod space D with respect to a modified skorohod metric, and h ranges over some compact metric space. Also, they proved that $n(\hat{\gamma} - \gamma_0)$ and $\sqrt{n}(\hat{\varphi} - \varphi_0)$ are asymptotically independent. For more details, see Kosorok and Song [2007] and Chapter 14 of Kosorok [2008b].

In the change-line regression problem, we have an additional change-line parameter α (angle) in addition to a cut-point γ. Note that we took middle points of sorted $\omega^T X_i, i = 1, \ldots, n$ to build a searching space for γ. Therefore, the angle α and the cut-point γ might not be independent. We explore the joint distribution of $(\hat{\alpha}, \hat{\gamma})$ obtained from simulations of the change-line regression models by using a graphical method.

6.2.1 Simulation set-up

We conducted 500 MC simulations to estimate both regression parameters (β, δ) and change-line parameters (α, γ) with different sample sizes 100, 200, and 300. True values for parameters were set to: $\beta_0 = (2,1)^T, \delta_0 = (-2,-1)$, $\omega_0 = (-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})^T$, equivalently $\alpha_0 = 2.356194$, and $\gamma_0 = 0$.

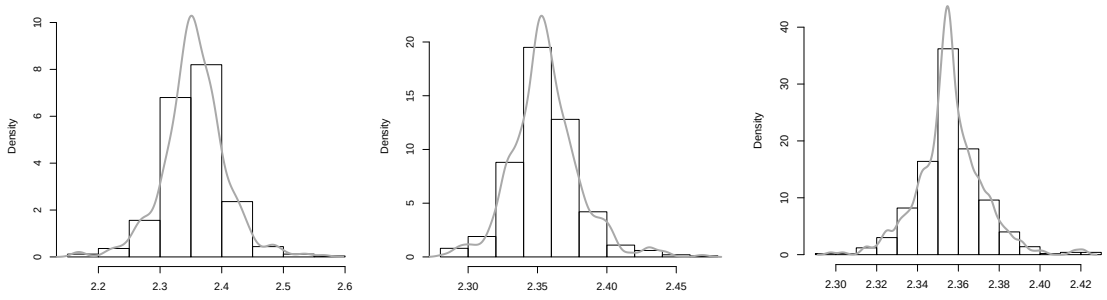
6.2.2 Preliminary results

1. Histogram and density plot of $\hat{\alpha}$ and $\hat{\gamma}$.

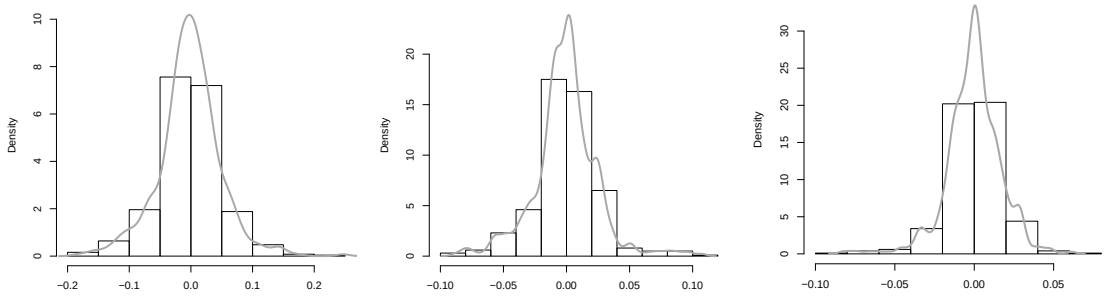
Figure 6.1 shows histogram and density plots of $\hat{\alpha}$ and $\hat{\gamma}$ based on 500 simulations for sample size of 100, 200, and 300. In this plot, densities were obtained by using a Gaussian kernel with a smoothing bandwidth implemented by the **R** function **density**. The estimates for angle and cut-points seem to be well-shaped, but are not very close to the shape of the normal distribution.

2. Graphical display of the empirical joint distribution of $\hat{\alpha}$ and $\hat{\gamma}$.

Next, we explored the empirical joint distribution of $\hat{\alpha}$ and $\hat{\gamma}$ through scatter plots and contour plots. Figure 6.2 displays scatter plots of $\hat{\alpha}$ against $\hat{\gamma}$ based on 500 simulations for sample sizes of 100, 200, and 300. Figure 6.3 shows the 2-dimensional contour plots of $\hat{\alpha}$ and $\hat{\gamma}$. Density was directly calculated by using 500 sets of $\hat{\alpha}$ and $\hat{\gamma}$ without Gaussian kernel smoothing. Figure 6.4 shows the same plot using a heat map. Figure 6.5 displays a 3-dimensional representation of the same contour plots. All scatter plots and contour plots suggest that $\hat{\alpha}$ and γ might be strongly associated, and appear to converge to a limiting process as sample size increases although we do not yet know the limiting distribution.

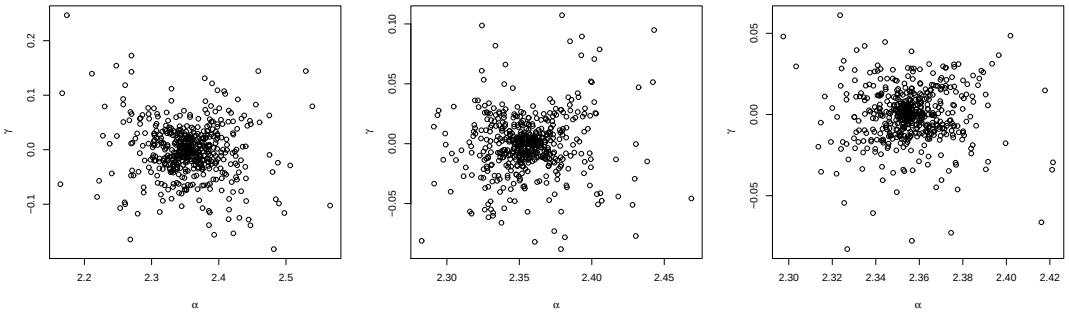


(a) $\hat{\alpha} : n = 100$ (b) $\hat{\alpha} : n = 200$ (c) $\hat{\alpha} : n = 300$



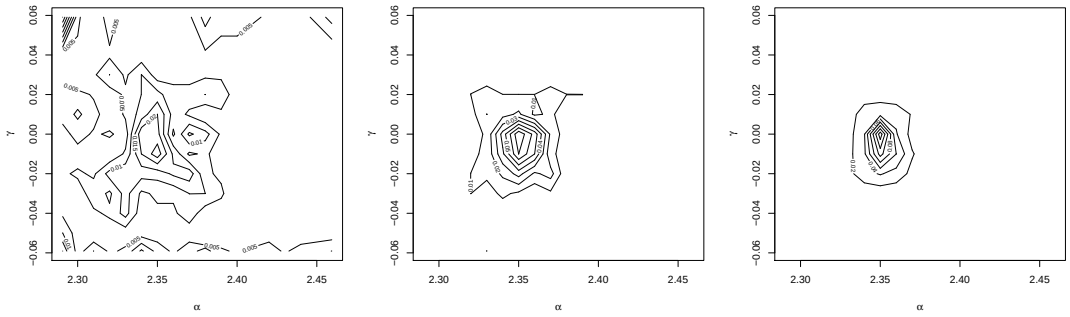
(d) $\hat{\gamma} : n = 100$ (e) $\hat{\gamma} : n = 200$ (f) $\hat{\gamma} : n = 300$

Figure 6.1: Histogram and density plot of $\hat{\alpha}$ and $\hat{\gamma}$ based on the 500 simulations for n=100, 200, and 300. Densities were calculated using the Gaussian kernel.



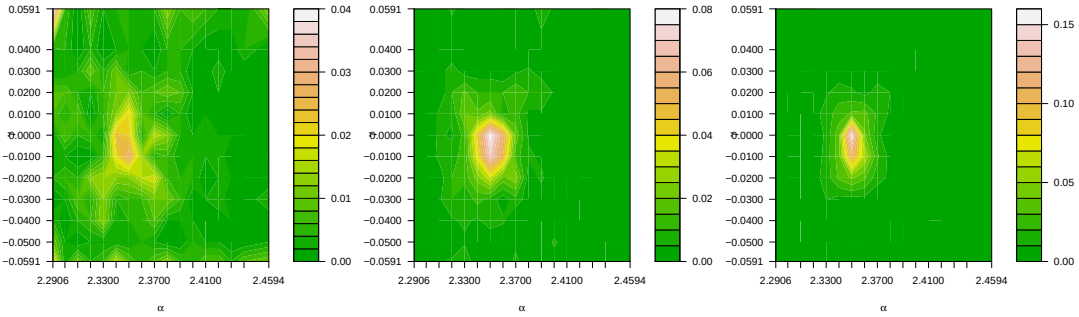
(a) $n = 100$ (b) $n = 200$ (c) $n = 300$

Figure 6.2: Scatter plots of $\hat{\alpha}$ against $\hat{\gamma}$ based on the 500 simulations for n=100, 200, and 300.



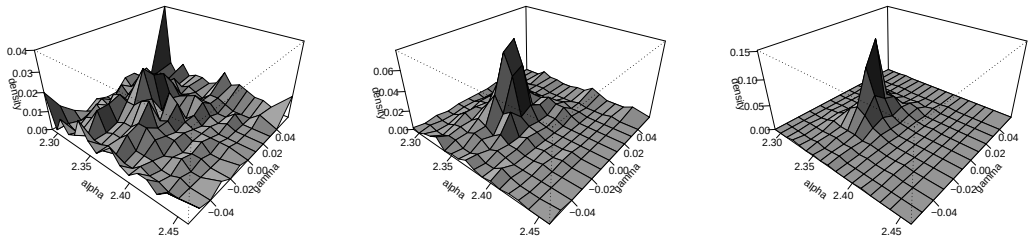
(a) $\hat{\alpha} : n = 100$ (b) $\hat{\alpha} : n = 200$ (c) $\hat{\alpha} : n = 300$

Figure 6.3: 2D contour plots of $\hat{\alpha}$ and $\hat{\gamma}$ based on the 500 simulations for n=100, 200, and 300. Densities were calculated based on the 500 sets of estimates.



(a) $n = 100$ (b) $n = 200$ (c) $n = 300$

Figure 6.4: 2D contour plots of $\hat{\alpha}$ and $\hat{\gamma}$ based on the 500 simulations for n=100, 200, and 300. Densities were calculated based on the 500 sets of estimates.



(a) $n = 100$ (b) $n = 200$ (c) $n = 300$

Figure 6.5: 3D contour plots of $\hat{\alpha}$ and $\hat{\gamma}$ based on the 500 simulations for n=100, 200, and 300. Densities were calculated based on the 500 sets of estimates.

Table 6.4: Validation of the rate of convergence for change-line parameters using 500 sets of $\hat{\alpha}$ and $\hat{\gamma}$.

sample size	Sample mean		Sample variance			
	$\hat{\alpha}$	$\hat{\gamma}$	$\hat{\alpha}$	ratio	$\hat{\gamma}$	ratio
100	2.3559	-0.002077	0.002578		0.002598	
200	2.3558	-0.000812	0.000591	4.3589	0.000649	3.99997
300	2.3560	-0.000424	0.000253	2.3351	0.000253	2.56386

Validation of the rate of convergence using sets of estimates

Table 6.4 displays sample means and variances of $\hat{\alpha}$ and $\hat{\gamma}$ over 500 simulations for sample sizes of 100, 200, and 300. Note that the rate of convergence for the change-line parameter is $\frac{1}{n}$, so we expect that the ratio of the sample variance of estimates in sample size 200 to the sample variance of estimates in sample size 100 is close to $\frac{2^2}{1} = 4$. Also, the ratio of the sample variance of estimates in sample size 300 to the sample variance of estimates in sample size 200 is close to $\frac{3^2}{2^2} = 2.25$. Columns 5 and 7 in Table 6.4 show the ratios of the estimates $\hat{\alpha}$ and $\hat{\gamma}$ respectively. The ratio of $\hat{\alpha}$ and $\hat{\gamma}$ in sample size 200 to those in sample size 100 are 4.3589 and 3.9999 respectively, which are quite close to the target value of 4. Also, the ratio of $\hat{\alpha}$ and $\hat{\gamma}$ in sample size 400 to those in sample size 200 are 2.3351 and 2.5639, respectively, which are also quite close to the target value of 2.25. Therefore, the rate of convergence theoretically calculated in the previous section can be verified empirically even for medium sample size.

6.3 The IDC method

In this dissertation, we proposed an interactive decision committee method that relies on different pairs of existing categories of feature variables as well as marginal feature categories and two-stage 5-fold cross-validation with forward selection. The IDC method was applied to two sets of chemical toxicity data from ToxRefDB and ICCVAM, consisting of binary endpoints and a set of feature variables from ten chemical descriptor blocks. Three learning algorithms were utilized as classification inducers: SVM, random forests, and tree-based AdaBoost. For simple comparison purposes, a stacked generalization with the IDC method as well as a single unaggregated classifier were applied to the same data set. The basic idea and computation of the IDC method is very simple, but the IDC method and the stacked generalization IDC method can improve prediction accuracies compared to a single classifier in the chemical toxicity data sets for all three learning algorithms.

Like other decision committee methods, the IDC method provides little insight into the decision-making process, and thus limited interpretation of the results could be made [Dietterich, 1997]. Despite this limitation, our work in this dissertation demonstrated that the proposed method can improve learning in the classification task, yielding higher prediction accuracy. This study suggests that the proposed IDC method with two-stage 5-fold CV and with stacked generalization could be useful to study classification problems when high-dimensional feature variables are grouped into feature categories. Also, the proposed method could be very useful in challenging QSAR classification problems, providing a useful tool for predicting hazards of chemicals, and prioritizing compounds for experimental assays.

6.4 Future research

6.4.1 Change-line classification and regression

This study is now in the early stages of development, and there are many possible ways to improve the proposed method. An important topic for future research is to extend and refine the split-line algorithm for feature space dimensions higher than two. We applied a probabilistic working model with the EM algorithm introduced in section 6.1.2 to the change-line classification problem with two heterogeneous subgroups determined by three-dimensional feature variables (the results are not presented in this paper). Three direction parameters ω_1, ω_2, and ω_3 can be expressed by using two angles $\theta \in (0, \pi)$ and $\psi \in (0, \pi)$, where $\omega_1 = \cos\theta$, $\omega_2 = \sin\theta\cos\psi$, and $\omega_3 = \sin\theta\sin\psi$ to satisfy the condition of $\|\omega\| = 1$. The estimated model parameters were quite close to the true values, but the direction parameters were not estimated well.

Also, this method seems to work very well when there are two latent subgroups, but it clearly needs to be extended to allow the possibility of more than two subgroups in the population. Several extensions can be made in model (3.1) to allow more than two subgroups, for example, by simply taking two different cut-points such as

$$Y(\theta; X, Z) \sim \mathbf{1}\{\omega^T X \le \gamma_1\} F(Z; \beta) + \mathbf{1}\{\gamma_1 < \omega^T X \le \gamma_2\} G(Z; \delta) + \mathbf{1}\{\omega^T X > \gamma_2\} H(Z; \alpha),$$

where $\omega \in \mathcal{S}^2, \gamma_1 < \gamma_2 \in [a, b] \in \mathbb{R}$, under the existing assumptions. To find γ_1, γ_2, we can use grid searching instead of line searching, which might be computationally more intensive.

One of the difficulties is how to handle large data sets such as the chemical toxicity example. To solve this problem, we can consider estimation based on subsampling

of observations frequently used in the machine learning approach. For the example of toxicity data analysis in this preliminary paper, 100 replications of subsampling for small portions of the whole sample were utilized, and this appeared to work well. Several positive aspects of inference based on subsampling include being able to approximate the correct limiting behavior. Unfortunately, the type of estimation approach used in this study has not been studied extensively. We have significant interest in estimation based on subsampling, and this will be one of our future research topics. Nevertheless, we very much need to develop more computationally efficient approaches to enumerating the relevant hyperplanes, and we plan on working earnestly on this problem in the future.

In addition, we are interested in developing a hypothesis test for the existence of a change-line. In this preliminary study, a graphic examination by Gaussian kernel estimation and local regression was performed to verify whether an abrupt change occurs in the mean and the variance of toxicity activity. There is extensive literature on hypothesis testing for the existence of a change-point based on the weighted bootstrap method (see, for example, Kosorok and Song [2007]) and based on subsampling (see, for example, Lee and Seo [2008]). We expect that a similar approach can be taken with hypothesis testing for the existence of a change-line in the two-dimensional feature space setting, and sup score test statistics and mean score test statistics using the bootstrap technique were examined in section 3.4. Due to computational difficulty, this investigation was carried out in a very limited setting. Therefore, further study to find a more computationally efficient method would be one of the future research topics related to the change-line problem.

We are also interested in additional asymptotic properties in the change-line regression model including weak convergence of the proposed M-estimators. Graphical investigation of the empirical distribution of the change-line parameters suggested that they might converge to a certain limiting processes, so establishing the asymptotic distribution of the estimators could be one of our future research topics. Although further research is needed to overcome its limitations, this preliminary study shows that the proposed method can be an attractive approach to finding latent subgroups in a population. Studying the asymptotic validation of the test procedure would be helpful.

6.4.2 The interactive decision committee method

The proposed IDC method brought up many open problems. First, the current IDC method can be extended to resolving multiclass classification problems [Hsu and Lin, 2002] or to predict continuous outcomes [Budka and Gabrys, 2010]. Second, it would be helpful to determine whether the improvement achieved by the IDC method in this paper can be observed in other types of data, such as gene expression data with gene categories. Liu et al. [2004] showed that a combination of feature selection with an ensemble neural network based on individual genes improved a classification task. Since we searched for all 2nd order interaction terms between feature categories, the current IDC method would be inefficient for a large number of categories. Gene pathways are numerous, so we would need a more efficient way to select 2nd order interaction terms between gene pathways. When feature categories can be defined in multiple ways, the best choice of feature categories is an open problem.

Finally, further studies to obtain more significant improvements using the IDC

method are needed. As several researchers including Breiman [2001] and Wolpert [1992] have argued, increasing diversity among base classifiers (or minimizing dependency or correlation between base classifiers) and improving performance of individual base classifiers are key factors in successful use of the decision committee method. Wang et al. [2009] reported that SVM with bagging or boosting performed better than a single SVM on average. Therefore, it would also be interesting to integrate bootstrap resampling techniques with the IDC method in order to increase diversity, thus potentially achieving better prediction performance similar to Assareh et al. [2008] and Stefanowski [2005]. One possible alternative is to use output class probabilities of the base classifiers rather than the predicted class levels of base classifiers as suggested in Ting et al. [1997]. However, Bauer and Kohavi [1999] argued that combining output class probabilities of the base classifier can produce slightly worse results than combining classification outputs, so this may not guarantee improved performance of the IDC method.

Chapter 7

Appendix

7.1 Empirical processes

Empirical processes are very useful to study asymptotic behavior of statistics. In this section, we introduce some major definitions, theories and lemmas that were used to study asymptotic properties in this thesis.

Definition 2. *[Kosorok, 2008b, p. 10] A class $\mathcal{F}$ of measurable functions $f : \mathcal{X} \mapsto \mathbb{R}$ is said to be a P-Glivenko-Cantelli (P-GC) class if $\sup_{f \in \mathcal{F}} |\mathbb{P}_n f - Pf| \to 0$ outer almost surely, where $\mathbb{P}_n f = n^{-1} \sum_{i=1}^{n} f(x_i)$ is the empirical measure for the sample $x_1, \ldots, x_n$, and $Pf = \int_{\mathcal{X}} f(x) P(dx)$.*

Definition 3. *[Kosorok, 2008b, p. 10] Define the random measure $\mathbb{G}_n = \sqrt{n}(\mathbb{P}_n - P)$, and for any class $\mathcal{F}$ of measurable functions $f : \mathcal{X} \mapsto \mathbb{R}$, let $\mathbb{G}$ be a mean zero Gaussian process indexed by $\mathcal{F}$, and with covariance $E[f(X)g(X)] - Ef(X)Eg(X)$ for all $f, g \in \mathcal{F}$, and having appropriately continuous sample path. We say that $\mathcal{F}$ is P-Donsker if $\mathbb{G}_n \rightsquigarrow \mathbb{G}$ in $\ell^\infty(\mathcal{F})$.*

Definition 4. *[Kosorok, 2008b, p. 156] Consider an arbitrary collection $\{x_1, \ldots, x_n\}$ of points in a set $\mathcal{X}$ and a collection $\mathcal{C}$ of subsets of $\mathcal{X}$. We say that $\mathcal{C}$ picks out a*

certain subset A of $\{x_1, \ldots, x_n\}$ if $A = C \cap \{x_1, \ldots, x_n\}$ for some $c \in \mathcal{C}$. We say that $\mathcal{C}$ shatters $\{x_1, \ldots, x_n\}$ if all of the 2^n possible subsets of $\{x_1, \ldots, x_n\}$ are picked out by the set in $\mathcal{C}$. The VC-index $V(\mathcal{C})$ of the class $\mathcal{C}$ is the smallest n for which no set of size n $\{x_1, \ldots, x_n\} \subset \mathcal{X}$ is shattered by $\mathcal{C}$. We say that $\mathcal{C}$ is a VC-class if $V(\mathcal{C}) < \infty$.

Definition 5. *[Kosorok, 2008b, p. 157] For a function $f : \mathcal{X} \mapsto \mathbb{R}$, the subset of $\mathcal{X} \times \mathbb{R}$ given by $\{(x,t) : t < f(x)\}$ is the subgraph of f. A collection $\mathcal{F}$ of measurable real functions on the sample space $\mathcal{X}$ is a VC-subgraph class or VC-class , if the collection of all subgraphs of functions in $\mathcal{F}$ forms a VC-class of sets. Let $V(\mathcal{F})$ denote the VC-index of the set of subgraphs of $\mathcal{F}$.*

Definition 6. *[Kosorok, 2008b, p. 18] For a class of functions $\mathcal{F}$ of functions $f : \mathcal{X} \mapsto \mathbb{R}$, $F : \mathcal{X} \mapsto \mathbb{R}$ is an "envelope" for $\mathcal{F}$ if $|f(x)| \leq F(x) < \infty$ for all $x \in \mathcal{X}$ and all $f \in \mathcal{F}$.*

Definition 7. *[Kosorok, 2008b, p. 142] A class $\mathcal{F}$ of measurable functions is pointwise measurable (PM) if there exists a countable subset $\mathcal{G} \subset \mathcal{F}$ such that for every $f \in \mathcal{F}$, there exists a sequence $\{g_m\} \in \mathcal{G}$ with $g_m(x) \to f(x)$ for every $x \in \mathcal{X}$.*

Definition 8. *[Kosorok, 2008b, p. 162] For a class of measurable functions $\mathcal{F}$, with envelope F, the uniform entropy integral*

$$J(\delta, \mathcal{F}, L_2) \equiv \int_0^\delta \sqrt{\sup_Q \log N(\varepsilon \|F\|_{Q,2}, \mathcal{F}, L_2(Q))} d\varepsilon,$$

where the supremum is taken over all finitely discrete probability measures Q with $\|F\|_{Q,2} > 0$. We say that the class $\mathcal{F}$ has bounded uniform entropy integral (BUEI) with envelope F if $J(1, \mathcal{F}, L_2) = \int_0^1 \sqrt{\sup_Q \log N(\varepsilon \|F\|_{Q,2}, \mathcal{F}, L_2(Q))} d\varepsilon < \infty$.

Definition 9. *[Van Der Vaart and Wellner, 1996, p. 52] For some metric $\mathbb{D}$, let $X_\alpha, X : \Omega \mapsto \mathbb{D}$ be arbitrary maps. Then, X_α converges almost surely to X if $P_*(\lim d(X_\alpha, X) = 0) = 1$.*

Definition 10. *[Kosorok, 2008b, p. 109] A sequence X_n is asymptotically tight if for every $\varepsilon > 0$, there is a compact K so that* $\liminf P_*(X_n \in K^\delta) \geq 1 - \varepsilon$*, for every $\delta > 0$, where for a set $A \subset \mathbb{D}$ and some metric space $\mathbb{D}$, $A^\delta = \{x \in \mathbb{D} : d(x, A) < \delta\}$ is the δ-enlargement around A.*

Lemma 1. *[Kosorok, 2008b, p. 142] (Lemma 8.10) Let $\mathcal{F}_1, \ldots, \mathcal{F}_k$ be PM classes of real functions on $\mathcal{X}$, and let $\phi : \mathbb{R}^k \mapsto \mathbb{R}$ be continuous. Then the class $\phi \circ (\mathcal{F}_1, \ldots, \mathcal{F}_k)$ is PM, where $\phi \circ (\mathcal{F}_1, \ldots, \mathcal{F}_k)$ denotes the class $\{\phi(f_1, \ldots, f_k) : (f_1, \ldots, f_k) \in \mathcal{F}_1 \times \ldots \times \mathcal{F}_k\}$.*

Proposition 1. *[Kosorok, 2008b, p. 143] (Proposition 8.11) Let $\mathcal{F}$ be a class of measurable functions $f : \mathcal{X} \mapsto \mathbb{R}$ on the probability space $(\mathcal{X}, \mathcal{A}, P)$. Provided $\mathcal{F}$ is PM with envelope F such that $P^*F^2 < \infty$, then $\mathcal{F}_\delta$ and $\mathcal{F}_\infty^2$ are PM for all $0 < \delta \leq \infty$.*

Lemma 2. *[Kosorok, 2008b, p. 143] (Lemma 8.12) Assume $\mathcal{F} \equiv \{\mathbf{1}\{Y - \beta' Z \leq t\} : \beta \in \mathbb{R}^k, t \in \mathbb{R}\}$. Then, the classes $\mathcal{F}, \mathcal{F}_\delta \equiv \{f - g : f, g \in \mathcal{F}, \|f - g\|_{P,2} < \delta\}$, and $\mathcal{F}_\delta^2 \equiv \{(f - g)^2 : f, g \in \mathcal{F}\}$ are all P-measurable for any probability measure on $\mathcal{X}$.*

Lemma 3. *[Kosorok, 2008b, p. 148] (Lemma 8.17) Let $\mathcal{F}$ be a class of measurable functions, with envelope $F \equiv \|f\|_{\mathcal{F}}$. For any $f, g \in \mathcal{F}$, define $\rho(f, g) \equiv \{P(f - Pf - g + Pg)^2\}^{1/2}$; and, for any $\delta > 0$, let $\mathcal{F}_\delta \equiv \{f - g : \rho(f, g) < \delta\}$. Then the following are equivalent:*

1. *$\mathcal{F}$ is P-Donsker.*
2. *$(\mathcal{F}, \rho)$ os totally bounded and $\|\mathbb{G}_n\|_{\mathcal{F}_{\delta_n}} \xrightarrow{P} 0$ in probability for every $\delta_n \downarrow 0$;*
3. *$(\mathcal{F}, \rho)$ os totally bounded and $E^*\|\mathbb{G}_n\|_{\mathcal{F}_{\delta_n}} \to 0$ for every $\delta_n \downarrow 0$.*

Theorem 3. *[Kosorok, 2008b, p. 149] (Theorem 8.19) Let $\mathcal{F}$ be a class of measurable functions with envelope F and the uniform entropy integral*

$$J(1,\mathcal{F},L_2)=\int_0^1\sqrt{\sup_Q \log N(\varepsilon\|F\|_{Q,2},\mathcal{F},L_2(Q))}d\varepsilon<\infty.$$

Let the classes $\mathcal{F}_\delta\equiv\{f-g:f,g\in\mathcal{F},\|f-g\|_{P,2}<\delta\}$, and $\mathcal{F}_\infty^2\equiv\{h^2:h\in\mathcal{F}\}$ be P-measurable (PM) for every $\delta>0$. If $P^*F^2<\infty$, then $\mathcal{F}$ is P-Donsker.

Theorem 4. *[Kosorok, 2008b, p. 157] (Theorem 9.2) There exists a universal constant $K<\infty$ such that, for a VC-class of sets $\mathcal{C}$, any $r\geq 1$, and any $0<\varepsilon<1$,*

$$N(\varepsilon,\mathbf{1}\{\mathcal{C}\},L_r(\mathbb{Q}))\leq KV(\mathcal{C})(4e)^{V(\mathcal{C})}(\frac{1}{\varepsilon})^{r(V(\mathcal{C}-1))}.$$

Theorem 5. *[Kosorok, 2008b, p. 157] (Theorem 9.3) There exists a universal constant $K<\infty$ such that, for any VC-class of measurable function $\mathcal{F}$ with integrable envelope function F, any $r\geq 1$, any probability measure $\mathbb{Q}$ with $\|F\|_{\mathbb{Q},r}>0$, and any $0<\varepsilon<1$,*

$$N(\varepsilon\|F\|_{\mathbb{Q},r},\mathcal{F},L_r(\mathbb{Q}))\leq KV(\mathcal{F})(4e)^{V(\mathcal{F})}(\frac{2}{\varepsilon})^{r(V(\mathcal{F}-1))}.$$

Lemma 4. *[Kosorok, 2008b, p. 159] (Lemma 9.6) Let $\mathcal{F}$ be a finite-dimensional vector space of measurable functions $f:\mathcal{X}\to\mathbb{R}$. Then $\mathcal{F}$ is VC-Subgraph with $V(\mathcal{F})\leq dim(\mathcal{F})+2$.*

Lemma 5. *[Kosorok, 2008b, p. 159] (Lemma 9.7) Let $\mathcal{C}$ and $\mathcal{D}$ be VC-classes of sets in a set χ, with respective VC-indices $V_\mathcal{C}$ and $V_\mathcal{D}$; and let $\mathcal{E}$ be a VC-class of sets in W, with VC-index $V_\mathcal{E}$. Also let $\phi:\chi\mapsto\mathcal{Y}$ and $\psi:\mathcal{Z}\mapsto\chi$ be fixed functions. Then*

1. *$\mathcal{C}^c\equiv\{C^c:C\in\}$ is VC with $V(\mathcal{C}^c)=V(\mathcal{C})$*

2. $\mathcal{C} \cap \mathcal{D} \equiv \{C \cap D : C \in \mathcal{C}, D \in \mathcal{D}\}$ *is VC with index* $\leq V_{\mathcal{C}} + V_{\mathcal{D}} - 1$;

3. $\mathcal{C} \cup \mathcal{D} \equiv \{C \cup D : C \in \mathcal{C}, D \in \mathcal{D}\}$ *is VC with index* $\leq V_{\mathcal{C}} + V_{\mathcal{D}} - 1$;

4. $\mathcal{D} \times \mathcal{E}$ *is VC in* $\chi \times W$ *with VC index* $\leq V_{\mathcal{D}} + V_{\mathcal{E}} - 1$;

5. $\phi(\mathcal{C})$ *is VC with index* $V_{\mathcal{C}}$ *if* ϕ *is one-to-one;*

6. $\psi^{-1}(\mathcal{C})$ *is VC with index* $\leq V_{\mathcal{C}}$.

Lemma 6. *[Kosorok, 2008b, p. 160] (Lemma 9.8) For any class* $\mathcal{C}$ *of sets in a set* $\mathcal{X}$*, the class* $\mathcal{F}_{\mathcal{C}}$ *of indicator functions of sets in* $\mathcal{C}$ *is VC-subgraph if and only if* $\mathcal{C}$ *is a VC-class. Moreover, whenever at least one of* $\mathcal{C}$ *or* $\mathcal{F}_{\mathcal{C}}$ *is VC, the respective VC-indices are equal.*

Lemma 7. *[Kosorok, 2008b, p. 160–161] (Lemma 9.9) Let* $\mathcal{F}$ *and* $\mathcal{G}$ *be VC-subgraph classes of functions on a set* $\mathcal{X}$*, with respective VC-indices* $V_{\mathcal{F}}$ *and* $V_{\mathcal{G}}$*. Let* $g : \mathcal{X} \mapsto \mathbb{R}$*,* $\phi : \mathbb{R} \mapsto \mathbb{R}$*, and* $\psi : \mathcal{Z} \mapsto \mathcal{X}$ *be fixed functions. Then,*

1. $\mathcal{F} \wedge \mathcal{G} \equiv \{f \wedge g : f \in \mathcal{F}, g \in \mathcal{G}\}$ *is VC-subgraph with index* $\leq V_{\mathcal{F}} + V_{\mathcal{G}} - 1$.

2. $\mathcal{F} \vee \mathcal{G}$ *is VC with index* $\leq V_{\mathcal{F}} + V_{\mathcal{G}} - 1$.

3. $\{\mathcal{F} > 0\} \equiv \{\{f > 0\} : f \in \mathcal{F}\}$ *is a VC-classes of sets with index* $V_{\mathcal{F}}$.

4. $-\mathcal{F}$ *is VC-subgraph with index* $V_{\mathcal{F}}$.

5. $\mathcal{F} + \mathcal{G} \equiv \{f + g : f \in \mathcal{F}\}$ *is VC with index* $V_{\mathcal{F}}$.

6. $\mathcal{F} \cdot g \equiv \{fg : f \in \mathcal{F}\}$ *is VC with index* $\leq 2V_{\mathcal{F}} - 1$.

7. $\mathcal{F} \circ \psi \equiv \{f(\psi) : f \in \mathcal{F}\}$ *is VC with index* $\leq V_{\mathcal{F}}$.

8. $\phi \circ \mathcal{F}$ *is VC with index* $\leq V_{\mathcal{F}}$ *for monotone* ϕ.

Lemma 8. *[Kosorok, 2008b, p. 161] (Lemma 9.12) The following are true:*

1. *The collection of all half-space in* $\mathbb{R}^d$*, consisting of the sets* $\{x \in \mathbb{R}^d :< x, u >\leq c\}$ *with* u *ranging over* $\mathbb{R}^d$ *and* c *ranging over* $\mathbb{R}$*, is VC with index* $d+2$*.*

2. *The collection of all closed balls in* $\mathbb{R}^d$ *is VC with index* $\leq d+3$*.*

Theorem 6. *(Argmax theorem) [Kosorok, 2008b, p. 265](Theorem 14.1) Let* M_n*,* M *be stochastic processes indexed by a metric space* H*. If the following conditions are satisfied,*

1. $M_n \rightsquigarrow M$ *in* $\ell^\infty(K)$ *for every compact* $K \subset H$
2. M *is an upper semicontinuous for almost all sample path* $h \mapsto M(h)$*.*
3. M *has a unique maximum at a (random) point* $\hat{h}$*.*
4. *a random map* $\hat{h}$ *is tight in* H*.*
5. $\hat{h}_n$ *is uniformly tight.*
6. $\hat{h}_n$ *satisfies* $M_n(\hat{h}_n) \geq \sup_{h \in H} M_n(h) - o_p(1)$*.*

Then $\hat{h}_n \rightsquigarrow \hat{h}$ *in* H*.*

Corollary 1. *[Kosorok, 2008b, p. 269](Corollary 14.5) Let* M_n *be a sequence of stochastic processes indexed by a semimetric* (Θ, d) *and* $M : \Theta \mapsto \mathbb{R}$ *a deterministic function. If the following conditions are satisfied:*

1. $M(\theta) - M(\theta_0) \leq -c_1 \tilde{d}^2(\theta, \theta_0)$*, where* $c_1 > 0$*,* $\tilde{d} : \Theta \times \Theta \mapsto [0, \infty)$ *satisfies* $\tilde{d}(\theta_n, \theta_0) \to 0$ *whenever* $d(\theta_n, \theta_0) \to 0$ *for every* θ *in a neighborhood of* θ_0*.*
2. $E^* \|\mathbb{G}_n\|_{\mathcal{M}_\delta} \lesssim \phi(\delta)$ *for a function* ϕ *such that* $\delta \mapsto \phi(\delta)/\delta^\alpha$ *is decreasing in* δ *for some* $\alpha < 2$*, and for every* n*.*
3. $\hat{\theta}_n$ *satisfies* $M_n(\hat{\theta}_n) \geq \sup_{\theta \in \Theta} M_n(\theta) - O_p(r_n^{-2})$ *and* $\hat{\theta}_n \to \theta_0$ *in outer probability.*
4. $r_n^2 \phi_n(r_n^{-1}) \leq c_3 \sqrt{(n)}$ *for every* n *and some* $c_3 \leq \infty$*.*

Then $r_n \tilde{d}(\hat{\theta}_n, \theta) = O_p(1)$*.*

7.2 Proof of consistency details

Proof of 2.1-A. The set $\{\omega^T X - \gamma > 0\}$ is a VC-class with VC-index ≤ 5.

As Van Der Vaart and Wellner [1996], p. 148) discussed, for any $f \in \mathcal{F}$, the sets $\{f > 0\}$ are a one-to-one image of the intersection of the subgraph with a set $\mathcal{X} \times \{0\}$. That is, $\{f > 0\}$ is equal to $\{(x_i, t_i) : f(x_i) > t_i\} \cap \mathcal{X} \times \{0\}$. Note that intersection of two VC-classes $C \cap D$ is also a VC-class with VC-index less than or equal to $V(C) + V(D) - 1$, where $V(C)$ and $V(D)$ are VC-indices for C and D, respectively. This is true because C can pick out at most $O(n^{V(C)-1})$ subsets, and D can pick out at most $O(n^{V(D)-1})$ subsets, so $C \cap D$ can pick out at most $O(n^{V(C)+V(D)-2}) < 2^n$ for large n. Also, the function $h : \mathcal{X} \times \mathbb{R} \mapsto \mathcal{X} \times 0$ defined by $h(x_i, t_i) = (x_i, 0)$ is one-to-one. Therefore, $\{(x_i, t_i) : f(x_i) > t_i\} \cap \mathcal{X} \times \{0\}$ is a VC-class, and hence $\{f > 0\}$ is a VC-class with VC-index $V(\mathcal{F})$. This verifies that the set $\{\omega' X - \gamma > 0\}$ is a VC-class with VC-index less than or equal to 5.

Proof of Claim 2.2. $\mathcal{F}$, $\mathcal{F}_\delta$, and $\mathcal{F}_\infty^2$ are P-measurable (PM) for every $\delta > 0$.

1. $\mathcal{F}$ is PM.

Consider $\mathcal{G} = \{1\{\omega^T X \leq \gamma\} : \omega \in \{(\omega_1, \omega_2) \in \mathbb{Q}^2, \|\omega\| = 1\}, x \in \mathcal{X}_M, \gamma \in \mathbb{Q}\}$ where $\mathbb{Q}$'s are the rationals. For a fixed $\omega \in \{(\omega_1, \omega_2) \in \mathbb{R}^2, \|\omega\| = 1\}$, and $\gamma \in \mathbb{R}$, we can construct a sequence $\{(\omega_m, \gamma_m)\}$ as follows: for each $m \geq 1$, choose $\omega_m \in \mathbb{Q}^2$ to satisfy $\|\omega_m - \omega\| \leq \frac{1}{2mM}$, and choose $\gamma_m \in \mathbb{Q}$ to satisfy $\gamma + \frac{1}{2m} < \gamma_m \leq \gamma + \frac{1}{m}$. Then, we can define g_m such as, for any $x \in \mathcal{X}_M$,

$$g_m = 1\{\omega_M^T X \leq \gamma_M\} = \{\omega^T X - \omega^T X + \omega_m^T X \leq \gamma_m + \gamma - \gamma\} = \{\omega^T X \leq \gamma + r_m\},$$

where $r_m = \gamma_m - \gamma + (\omega - \omega_m)^T X$. Since $\frac{1}{2m} < \gamma_m - \gamma \leq \frac{1}{m}$, and $-\frac{1}{2m} < (\omega - \omega_m)^T X < \frac{1}{2m}$, $0 < r_m = (\gamma_m - \gamma) + (\omega - \omega_m)^T X < \frac{3}{2m}$ for all $m \geq 1$. We can see that $r_m \to 0$ as $m \to \infty$

since $\gamma_m \to \gamma$ and $\|\omega_m - \omega\| \le \frac{1}{2mM} \to 0$ when $\|X\| \le M < \infty$ as $m \to \infty$. Note that when $\mathbf{1}\{\omega^T X \le \gamma\}$ is right-continuous and X is arbitrary, $\lim_{r_m \to 0} \mathbf{1}\{\omega^T X \le \gamma + r_m\} = \mathbf{1}\{\omega^T X \le \gamma\}$, and hence $g_m = \mathbf{1}\{\omega^T X \le \gamma + r_m\} \to f = \mathbf{1}\{\omega^T X \le \gamma\}$ for all $X \in \mathcal{X}_M$. Therefore, we conclude that $\mathcal{F}$ is PM.

2. $\mathcal{F}_\delta = \{f_1 - f_2 : f_1, f_2 \in \mathcal{F}, \|f_1 - f_2\|_{P,2} < \delta\}$ **is PM.**

Again, we consider $\mathcal{F}_\delta$ in $\mathcal{X}_M$. Similar to 1, we can construct sequences $\{g_{m,1}\}, \{g_{m,2}\} \in \mathcal{G}$ such that, for each $m \ge 1$, $g_{m,1} = \mathbf{1}\{\omega^T X \le \gamma + r_{m,1}\}$, $g_{m,2} = \mathbf{1}\{\omega^T X \le \gamma + r_{m,2}\}$, satisfies $r_{m,i} > 0$, and $r_{m,i} \to 0$, as $m \to \infty$, for both $i = 1, 2$. Then, we have $g_{m,i} \to f_i$ for $i = 1, 2$. To show $\|g_{m,1} - g_{m,2}\|_{P,2} \to \|f_1 - f_2\|_{P,2}$, we can use the Dominated convergence theorem (DCT) as follows. First, $|g_{m,i}| \le 1$ for $i = 1, 2$, and 1 is integrable on $\mathcal{X}_M$. Thus $\|g_{m,1} - f_1\|_{P,1} \to 0$ and $\|g_{m,2} - f_2\|_{P,1} \to 0$ as $m \to \infty$ by DCT. Next, let Y_m denote $g_{m,1} - g_{m,2}$, and Y denote $f_1 - f_2$. Note that

$$\begin{aligned} |Y_m^2| &= |(g_{m,1} - g_{m,2})^2| \\ &= |\mathbf{1}\{\omega^T X \le \gamma + r_{m,1}\} + \mathbf{1}\{\omega^T X \le \gamma + r_{m,2}\} - 2 \times \mathbf{1}\{\omega^T X \le \gamma + r_{m,1}\}\mathbf{1}\{\omega^T X \le \gamma + r_{m,2}\}| \\ &\le 2, \end{aligned}$$

and 2 is integrable on $\mathcal{X}_M$. Now, for any $\varepsilon > 0$,

$$\begin{aligned} P(|Ym - Y| > \varepsilon) &= P(|(g_{m,1} - f_1) + (f_2 - g_{m,2})| > \varepsilon) \\ &\le \frac{P(|g_{m,1} - f_1| + |g_{m,2} - f_2|)}{\varepsilon} \\ &\le \frac{P(|g_{m,1} - f_1|) + P(|g_{m,2} - f_2|)}{\varepsilon} \to 0, \qquad \text{as } m \to \infty. \end{aligned}$$

Therefore, $Y_m \xrightarrow{p} Y$, and hence $Y_m^2 \xrightarrow{p} Y^2$ by the continuous mapping theorem. Combining these two previous properties, we have $|Y_m^2| \le 2$ and $Y_m^2 \xrightarrow{p} Y^2$, so we can

conclude that $\int Y_m^2 dp \to \int Y^2 dp$, that is, $\|g_{m,1} - g_{m,2}\|_{P,2} \to \|f_1 - f_2\|_{P,2}$.

3. $\mathcal{F}_\infty^2 = \{h^2 : h \in \mathcal{F}\}$ is PM.

We follow the proof of Lemma 8.10 [Kosorok, 2008b, p. 142], with $\phi(x) = x^2$. Let $\mathcal{H} = \phi(\mathcal{F}_1, \ldots, \mathcal{F}_K)$, and it suffices to show that there exists a countable subset $\mathcal{G}^* \subset \mathcal{H}$ with $\{g_m^*\} \in \mathcal{G}^*$ satisfying $g_m^* \to h$ for any $h \in \mathcal{H}$. Note that each $\mathcal{F}_i$ has a countable subset $\mathcal{G}_i \subset \mathcal{F}_i$ with a subsequence $\{g_m^i\} \in \mathcal{G}_i$ satisfying $g_m^i(x) \to f_i(x)$ as $m \to \infty$ for all $x \in \mathcal{X}$ and $i = 1, 2, \ldots, K$ when $\mathcal{F}$ is PM. Since ϕ is a continuous function, $\phi(x_n) \xrightarrow{p} \phi(x)$ whenever $x_n \xrightarrow{p} x$. Therefore, $\phi(g_m^1, \ldots, g_m^K)(x) \to \phi(f_1, \ldots, f_K) = h(x)$ as $m \to \infty$. Since h is arbitrary in $\mathcal{H}$, we can conclude that $\mathcal{G}^* = \phi(\mathcal{G}_1, \ldots, \mathcal{G}_K)$ is a countable subset of $\mathcal{H}$, making $\mathcal{H} = \mathcal{F}_\infty^2$ PM.

Bibliography

Abeel, T., Helleputte, T., Van de Peer, Y., Dupont, P., and Saeys, Y. (2009). Robust biomarker identification for cancer diagnosis using ensemble feature selection methods. *Bioinformatics*, 23(19).

Ali, K. and Pazzani, M. (1996). Error reduction through learning multiple descriptions. *Machine Learning*, 24(3):173–202.

Andrews, D. (2001). Testing when a parameter is on the boundary of the maintained hypothesis. *Econometrica*, 69(3):683–734.

Andrews, D. and Ploberger, W. (1994). Optimal tests when a nuisance parameter is present only under the alternative. *Econometrica: Journal of the Econometric Society*, pages 1383–1414.

Ashburner, M., Ball, C., Blake, J., Botstein, D., Butler, H., Cherry, J., Davis, A., Dolinski, K., Dwight, S., Eppig, J., et al. (2000). Gene ontology: tool for the unification of biology. *Nature genetics*, 25(1):25–29.

Assareh, A., Moradi, M., and Volkert, L. (2008). A hybrid random subspace classifier fusion approach for protein mass spectra classification. *Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics*, pages 1–11.

Banerjee, M. and McKeague, I. (2007). Confidence sets for split points in decision trees. *The Annals of Statistics*, 35(2):543–574.

Bauer, E. and Kohavi, R. (1999). An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. *Machine learning*, 36(1):105–139.

Breiman, L. (1996). Bagging predictors. *Machine learning*, 24(2):123–140.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Breiman, L., Friedman, J., Olshen, R., and Stone, C. (1984). Classification and regression trees. Wadsworth & Brooks. *Cole, Pacific Grove, California, USA*.

Budka, M. and Gabrys, B. (2010). Ridge regression ensemble for toxicity prediction. *Procedia Computer Science*, 1(1):193–201.

Chang, C.-C. and Lin, C.-J. (2001). *LIBSVM: a library for support vector machines*. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Cherkasov, A. (2005). Inductive QSAR descriptors. Distinguishing compounds with antibacterial activity by artificial neural networks. *Int. J. Mol. Sci*, 6:63–86.

Chernoff, H. and Zacks, S. (1964). Estimating the current mean of a normal distribution which is subjected to changes in time. *The Annals of Mathematical Statistics*, 35(3):999–1018.

Clemen, R. (1989). Combining forecasts: A review and annotated bibliography. *International Journal of Forecasting*, 5(4):559–583.

Cleveland, W. (1979). Robust locally weighted regression and smoothing scatterplots. *Journal of the American statistical association*, 74(368):829–836.

database, C. (2008). National Library of MEdicine.

DAVTES, R. (1977). Hypothesis testing when a nuisance parameter is present only under the alternative. *Biometrika*, 64(2):247.

Dietterich, T. (1997). Machine-learning research. *AI magazine*, 18(4):97.

DRAGON (2006). *DRAGON for Windows (Software for Molecular Descriptor Calculations). Talete s.r.1., Milan (Italy).*

Dutta, D., Guha, R., Wild, D., and Chen, T. (2007). Ensemble feature selection: consistent descriptor subsets for multiple QSAR models. *J. Chem. Inf. Model*, 47(3):989–997.

Freund, Y. and Schapire, R. (1997). A desicion-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55:119–139.

Friedman, J. and Stuetzle, W. (1980). Projection pursuit classification. *Unpublished manuscript*.

Friedman, J. and Stuetzle, W. (1981). Projection pursuit regression. *Journal of the American statistical Association*, 76(376):817–823.

Friedman, J., Stuetzle, W., and Schroeder, A. (1984). Projection pursuit density estimation. *Journal of the American Statistical Association*, 79(387):599–608.

Friedman, J. and Tukey, J. (1974). A projection pursuit algorithm for exploratory data analysis. *Computers, IEEE Transactions on*, 100(9):881–890.

Glynn, S., Boersma, B., Howe, T., Edvardsen, H., Geisler, S., Goodman, J., Ridnour, L., Lønning, P., Børresen-Dale, A., Naume, B., et al. (2009). A mitochondrial target sequence polymorphism in manganese superoxide dismutase predicts inferior survival in breast cancer patients treated with cyclophosphamide. *Clinical Cancer Research*, 15(12):4165.

Guo, J., Wall, M., and Amemiya, Y. (2006). Latent class regression on latent factors. *Biostatistics*, 7(1):145.

Hansen, L. and Salamon, P. (1990). Neural network ensembles. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 12(10):993–1001.

Hartigan, J. (1994). Linear estimators in change point problems. *The Annals of Statistics*, 22(2):824–834.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The elements of statistical learning: data mining, inference, and prediction*. Springer Verlag.

Ho, T. (1998). The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):832–844.

Hsu, C. and Lin, C. (2002). A comparison of methods for multiclass support vector machines. *Neural Networks, IEEE Transactions on*, 13(2):415–425.

Huber, P. (1985). Projection pursuit. *The annals of Statistics*, pages 435–475.

Johnson, S. (2008). The trouble with QSAR (or how I learned to stop worrying and embrace fallacy). *J. Chem. Inf. Model*, 48(1):25–26.

Karatzoglou, A., Meyer, D., and Hornik, K. (2006). Support vector machines in R. *Journal of Statistical Software*, 15(9):1–28.

Karlis, D. and Xekalaki, E. (2003). Choosing initial values for the em algorithm for finite mixtures. *Computational Statistics & Data Analysis*, 41(3-4):577–590.

Kosorok, M. (2008a). Dividing a point cloud in two dimensionals using lines. *Unpublished manuscript*.

Kosorok, M. (2008b). *Introduction to empirical processes and semiparametric inference*. Springer Verlag.

Kosorok, M. and Song, R. (2007). Inference under right censoring for transformation models with a change-point based on a covariate threshold. *The Annals of Statistics*, 35(3):957–989.

Krogh, A. and Vedelsby, J. (1995). Neural network ensembles, cross validation, and active learning. *Advances in neural information processing systems*, pages 231–238.

Kruskal, J. (1969). Toward a practical method which helps uncover the structure of a set of multivariate observations by finding the linear transformation which optimizes a new index of condensation. In *Statistical Computation*, pages 427–440.

Kruskal, J. (1972). Linear transformation of multivariate data to reveal clustering. *Multidimensional Scaling: Theory and Application in the Behavioral Sciences, I, Theory*, pages 179–191.

Kuncheva, L., Bezdek, J., and Duin, R. (2001). Decision templates for multiple classifier fusion: an experimental comparison. *Pattern Recognition*, 34(2):299–314.

Lam, L. (2000). Classifier combinations: implementations and theoretical issues. *Multiple classifier systems*, pages 77–86.

Langseth, H. and Nielsen, T. (2005). Latent classification models. *Machine learning*, 59(3):237–265.

Langseth, H. and Nielsen, T. (2009). Latent classification models for binary data. *Pattern Recognition*, 42(11):2724–2736.

Lee, E., Cook, D., Klinke, S., and Lumley, T. (2005). Projection pursuit for exploratory supervised classification. *Journal of Computational and Graphical Statistics*, 14(4):831–846.

Lee, S. and Seo, M. (2008). Semiparametric estimation of a binary response model with a change-point due to a covariate threshold. *Journal of Econometrics*, 144(2):492–499.

Liaw, A. and Wiener, M. (2002). Classification and regression by randomforest. *R News*, 2(3):18–22.

Lienemann, K., Plötz, T., and Fink, G. (2007). On the application of SVM-Ensembles based on adapted random subspace sampling for automatic classification of NMR data. *Multiple Classifier Systems*, pages 42–51.

Lin, C., Moré, J., et al. (1999). Newton's method for large bound-constrained optimization problems. *SIAM Journal on Optimization*, 9(4):1100–1127.

Liu, B., Cui, Q., Jiang, T., and Ma, S. (2004). A combinational feature selection and ensemble neural network method for classification of gene expression data. *BMC bioinformatics*, 5(1):136.

Malzahn, D. and Opper, M. (2003). An approximate analytical approach to resampling averages. *The Journal of Machine Learning Research*, 4:1151–1173.

Mangasarian, O. and Musicant, D. (1999). Successive overrelaxation for support vector machines. *IEEE Transactions on Neural Networks*, 10(5):1032–1037.

Manski, C. (1975). Maximum score estimation of the stochastic utility model of choice. *Journal of Econometrics*, 3(3):205–228.

Manski, C. (1985). Semiparametric analysis of discrete response:: Asymptotic properties of the maximum score estimator. *Journal of Econometrics*, 27(3):313–333.

Martin, M., Judson, R., Reif, D., Kavlock, R., and Dix, D. (2009a). Profiling chemicals based on chronic toxicity results from the US EPA ToxRef Database. *Environmental health perspectives*, 117(3):392.

Martin, M., Mendez, E., Corum, D., Judson, R., Kavlock, R., Rotroff, D., and Dix, D. (2009b). Profiling the reproductive toxicity of chemicals from multigeneration studies in the Toxicity Reference Database (ToxRefDB). *Toxicological Sciences*.

Martin, T., Harten, P., Venkatapathy, R., Das, S., and Young, D. (2008). A hierarchical clustering methodology for the estimation of toxicity. *Toxicology Mechanisms and Methods*, 18(2-3):251–266.

Mazzatorta, P., Cronin, M., and Benfenati, E. (2006). A QSAR study of avian oral toxicity using support vector machines and genetic algorithms. *QSAR & Combinatorial Science*, 25(7):616–628.

Muller, H. (1992). Change-points in nonparametric regression analysis. *The Annals of Statistics*, 20(2):737–761.

Nader, I., Tran, U., and Formann, A. (2011). Sensitivity to initial values in full nonparametric maximum-likelihood estimation of the two-parameter logistic model. *British Journal of Mathematical and Statistical Psychology*, 64(2):320–336.

NIEHS (2009). Interagency Coordinating Committee on the Validation of Alternative Methods. 2009. ICCVAM Test Method Evaluation Report. The Reduced Murine Local Lymph Node Assay: An Alternative Test Method Using Fewer Animals to Assess the Allergic Contact Dermatitis Potential of Chemicals and Products. NIH Publication Number 09-6439. Research Triangle Park, NC: National Institute of Environmental Health Sciences.

Ninomiya, Y. (2004). Construction of conservative test for change-point problem in two-dimensional random fields. *Journal of multivariate analysis*, 89(2):219–242.

Opitz, D. (1999). Feature selection for ensembles. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI)*, pages 379–384. John Wiley & Sons LTD.

Opitz, D. and Maclin, R. (1999). Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research*, 11(1):169–198.

Page, E. (1954). Continuous inspection schemes. *Biometrika*, 41(1/2):100–115.

Park, M. and Hastie, T. (2008). Penalized logistic regression for detecting gene interactions. *Biostatistics*, 9(1):30.

Pastor, R. and Guallar, E. (1998). Use of two-segmented logistic regression to estimate change-points in epidemiologic studies. *American journal of epidemiology*, 148(7):631.

Pastor-Barriuso, R., Guallar, E., and Coresh, J. (2003). Transition models for change-point estimation in logistic regression. *Statistics in medicine*, 22(7):1141–1162.

Polishchuk, P., Muratov, E., Artemenko, A., Kolumbin, O., Muratov, N., and Kuzmin, V. (2009). Application of random forest approach to QSAR prediction of aquatic toxicity. *Journal of chemical information and modeling*, 49(11):2481–2488.

Pons, O. (2003). Estimation in a Cox regression model with a change-point according to a threshold in a covariate. *The Annals of Statistics*, 31(2):442–463.

R Development Core Team (2010). *R: A Language and Environment for Statistical Computing*. Vienna, Austria. ISBN 3-900051-07-0.

Raftery, A. (1994). Change point and change curve modeling in stochastic processes and spatial statistics. *Journal of Applied Statistical Science*, 1(4):403–423.

Richard, A. (2006). Future of Toxicology Predictive Toxicology: An Expanded View of HChemical Toxicity. *Chemical research in toxicology*, 19(10):1257–1262.

Ripley, B. (2008). *Pattern recognition and neural networks*. Cambridge Univ Pr.

Shin, H. and Markey, M. (2006). A machine learning perspective on the development of clinical decision support systems utilizing mass spectra of blood samples. *Journal of Biomedical Informatics*, 39(2):227–248.

Shipp, C. and Kuncheva, L. (2002). Relationships between combination methods and measures of diversity in combining classifiers. *Information Fusion*, 3(2):135–148.

Sigletos, G., Paliouras, G., Spyropoulos, C., and Hatzopoulos, M. (2005). Combining information extraction systems using voting and stacked generalization. *The Journal of Machine Learning Research*, 6:1751–1782.

Silverman, B. (1986). *Density estimation for statistics and data analysis*, volume 26. Chapman & Hall/CRC.

Stefanowski, J. (2005). An experimental study of methods combining multiple classifiers-diversified both by feature selection and bootstrap sampling. *Issues in the Representation and Processing of Uncertain and Imprecise Information, Akademicka Oficyna Wydawnicza EXIT, Warszawa*, pages 337–354.

Stouch, T., Kenyon, J., Johnson, S., Chen, X., Doweyko, A., and Li, Y. (2003). In silico ADME/Tox: why models fail. *Journal of computer-aided molecular design*, 17(2):83–92.

Su, X., Tsai, C., Wang, H., Nickerson, D., and Li, B. (2009). Subgroup analysis via recursive partitioning. *The Journal of Machine Learning Research*, 10:141–158.

Thurston, R., Matthews, K., Hernandez, J., and De La Torre, F. (2009). Improving the performance of physiologic hot flash measures with support vector machines. *Psychophysiology*, 46(2):285–292.

Ting, K., Witten, I., and of Waikato. Dept. of Computer Science, U. (1997). *Stacked Generalization: when does it work?* Citeseer.

Tuv, E., Borisov, A., Runger, G., and Torkkola, K. (2009). Feature selection with ensembles, artificial variables, and redundancy elimination. *The Journal of Machine Learning Research*, 10:1341–1366.

Vale, K., Dias, F., Canuto, A., et al. (2008). A class-based feature selection method for ensemble systems. In *Eighth International Conference on Hybrid Intelligent Systems*, pages 596–601. IEEE.

van der Heijden, F., Duin, R., De Ridder, D., and Tax, D. (2004). *Classification, parameter estimation and state estimation*. Wiley Online Library.

Van Der Vaart, A. and Wellner, J. (1996). *Weak convergence and empirical processes*. Springer Verlag.

Vapnik, V., Golowich, S., and Smola, A. (1996). Support vector method for function approximation, regression estimation, and signal processing. In *Advances in Neural Information Processing Systems 9*. Citeseer.

Vapnik, V. N. (1998). *Statistical learning theory*, volume 2. Wiley New York.

Wang, S., Mathew, A., Chen, Y., Xi, L., Ma, L., and Lee, J. (2009). Empirical analysis of support vector machine ensemble classifiers. *Expert Systems with Applications*, 36(3):6466–6476.

Wolpert, D. (1992). Stacked generalization*. *Neural networks*, 5(2):241–259.

Wu, J. and Chu, C. (1993). Kernel-type estimators of jump points and values of a regression function. *The Annals of Statistics*, 21(3):1545–1566.

Yu, W., Liu, T., Valdez, R., Gwinn, M., and Khoury, M. (2010). Application of support vector machine modeling for prediction of common diseases: the case of diabetes and pre-diabetes. *BMC Medical Informatics and Decision Making*, 10(1):16.

Zhao, C., Zhang, H., Zhang, X., Liu, M., Hu, Z., and Fan, B. (2006). Application of support vector machine (SVM) for prediction toxic activity of different data sets. *Toxicology*, 217(2-3):105–119.

Zhu, H., Martin, T., Ye, L., Sedykh, A., Young, D., and Tropsha, A. (2009). Quantitative Structure- Activity Relationship Modeling of Rat Acute Toxicity by Oral Exposure. *Chemical research in toxicology*, 22(12):1913–1921.

Zhu, H., Tropsha, A., Fourches, D., Varnek, A., Papa, E., Gramatica, P., Oberg, T., Dao, P., Cherkasov, A., and Tetko, I. (2008). Combinatorial QSAR modeling of chemical toxicants tested against Tetrahymena pyriformis. *Journal of chemical information and modeling*, 48(4):766–784.

Zhu, M. (2008). Kernels and ensembles. *The American Statistician*, 62(2):97–109.

Zou, F., Lee, S., Knowlesc, M., and Wright, F. (2009). Quantification of population structure using correlated SNPs by shrinkage principal components. *Submitted*.